NTNU – Trondheim
Norwegian University of
Science and Technology

Isogeometric analysis of acoustic scattering

Jon Vegard Venås

Master of Science in Mathematics
Submission date: June 2015
Supervisor: Trond Kvamsdal, MATH
Co-supervisor: Trond Jenserud, FFI

Norwegian University of Science and Technology
Department of Mathematical Sciences

Summary

Acoustic scattering has been thoroughly analyzed with the use of finite element analysis (FEA). The problem at hand is a coupled fluid-structure interaction problem on an unbounded domain, where an object of elastic material is surrounded by fluid. Using physical assumptions, the fluid is described by the wave equation which is transformed to the Helmholtz equation. That is, the frequency domain is considered instead of the time domain. In particular one is interested in the scattered pressure of a plane wave at a far field point. The far field is computed by first computing the near field of the fluid surrounding the object, and integrating this solution together with Green's function to compute the scattered pressure at a far field point. This results in the so-called target strength (TS) of an object. This is done for a series of frequencies in the lower spectrum, such that the TS may be plotted over the frequency spectrum. It will also be possible to plot the TS for a range of different incident plane waves, in addition to measuring the TS for a range of angles around the scatterer. As the domain is unbounded, one must create an artificial boundary in order to use FEA. There exist several methods for ensuring that the outgoing waves are not reflected at this artificial boundary, including the PML method after Bérenger, several absorbing boundary condition operators (ABC-operators) after Enquist and Majda (among others), the use of Dirichlet-to-Neumann (DtN) operators and finally the use of so-called infinite elements. A complete presentation of this analysis of acoustic scattering with the use of infinite elements has been presented by Frank Ihlenburg. The same analysis using isogeometric analysis (IGA), will be presented. An overview of IGA alongside the physical equations will be presented for completeness. A complete program will be developed in MATLAB, and GLview Inova will be used as a supplementary program for post processing.

The elastic object at hand is exactly represented by NURBS, and is thus especially suited for IGA. Moreover, it has been shown that IGA is superior to the FEA when considering spectrum analysis.

An extensive analysis of the spherical shell, where the 3D solution is known, will be presented. First, the eigenvalues for the elastic spherical shell is computed, then the acoustic scattering on a rigid sphere is analyzed and finally the full problem is analyzed where acoustic scattering of an elastic spherical shell is considered. All of this is computed using IGA and compared to the known analytic solutions.

The workshop Benchmark Target Strength Simulation (BeTSSi) has been a forum for the problem at hand, where several benchmark object has been analyzed by FEA and other methods. The complexity of these objects varies all the way up to a full submarine including torpedoes. Common for all objects is that no analytic solutions are found. Comparisons using several methods have been presented in the BeTSSi conference, with deviating results. The resulting TS graphs from IGA will be compared to existing methods.

Sammendrag

Akustisk spredning har blitt grundig analysert med bruk av endelig element analyse (FEA). Problemstillingen er et koblet interaksjonsproblem mellom et fluid og et elastisk materiale, hvor det elastiske materialet er omgitt av fluidet. Med noen fysiske antagelser beskrives fluidet med bølgeligningen som transformeres til Helmholtz ligning. Altså betraktes frekvensdomenet istedenfor tidsdomenet. Man er spesielt interessert i det spredte trykket langt unna objektet. Dette beregnes ved å finne løsningen nært objektet, for så å integrere denne løsningen sammen med Green's funksjon. Dette resulterer i den såkalte målstyrken (TS) til objektet. Denne beregningen blir gjort for en rekke frekvenser i det nedre spektrum, slik at man kan plotte TS mot frekvensdomenet. Det er også mulig å plotte TS for forskjellige innfallende plane bølger, samt å plotte TS for forskjellige punkter rundt sprederen. Siden domenet er ubegrenset må man lage en kunstig rand for å kunne bruke FEA. Det eksisterer flere metoder for å sikre at ingen utgående bølger reflekteres fra denne kunstige randen. Dette inkluderer PML metoden til Bérenger, flere absorberende randebetingelser (ABC-operatorer) etter Enquist og Majda (blant flere), bruken av Dirichlet-til-Neumann (DtN) operatorer og de såkalte uendelige elementene. En komplett analyse av akustisk refleksjon med bruk av uendelige elementer har blitt presentert av Frank Ihlenburg. Den samme analysen hvor isogeometrisk analyse (IGA) brukes vil her bli presentert. En introduksjon til IGA sammen med de fysiske ligningene vil også bli presentert. Det vil bli utviklet et komplett program i MATLAB, og GLview Inova vil bli brukt til som et komplimenterende postprosesseringsprogram.

Det elastiske objektet representeres med NURBS, og er dermed spesielt egnet for IGA. Videre har det blitt vist at IGA er bedre enn FEA for spektrum analyse. En grundig analyse av et sfærisk skall, hvor den fulle 3D løsningen er kjent, vil bli presentert. Først beregnes egenverdiene for det sfæriske skallet (som er elastisk), så analyseres akustisk spredning på en fast sfære og til slutt analyseres akustisk spredning på det sfæriske skallet. Alt dette blir beregnet med IGA og sammenlignet med kjente analytiske løsninger og verdier.

Benchmark Target Strength Simulation (BeTSSi) er et forum for problemstillingen hvor flere standardmodeller har blitt analysert av FEA og andre metoder. Kompleksiteten av disse modellene varierer helt opp til en full ubåt med indre strukturer som for eksempel torpedoer. Felles for alle objekter er at ingen analytisk løsning er kjent. En sammenligning av forskjellige metoder anvendt på disse modellene har blitt presentert på BeTSSi konferansene, med avvikende resultater. Resultatene fra IGA vil bli presentert opp mot eksisterende resultater.

Contents

List of Figures

List of Tables

Preface

This thesis concludes my master's degree in Mathematics at the Norwegian University of Technology and Science (NTNU). The work spans both the fall 2014 and the spring 2015. The thesis represents the second half of the 2 year master program, which is a continuation of my bachelor's in degree at NTNU.

The thesis was proposed by my supervisor Trond Jenserud[1]. At that time, I had experience both with FEM and IGA as I had been following three courses involving these methods. These courses would set the foundation of the knowledge needed to complete a thesis on the subject. All of these courses was supervised by Trond Kvamsdal[2], and it was thus natural to choose Kvamsdal as an adviser at NTNU. Although two complete programs had been made for FEM and IGA (both involving solution of elasticity problems) in these courses, I decided to start from scratch by building each part of the code in a structural way. This way, I was able to create generic functions which could solve a larger set of problems. To test the validity of the implementations, many test examples was executed, mostly from the article and book of Hughes et al. ([1] and [2]). Some additional test examples were invented to validate special aspects of the implementations. These include linear elasticity problem on a Solid circular cylinder and rectangular prisms. This groundwork then enabled more complicated simulations of the vibration analysis, exterior Helmholtz problems and finally fully structure interaction problems.

The work of this thesis has been to consider the application of IGA. After all, the chosen specialization in the master's degree is applied mathematics. One typically presents the foundation of the theory from functional analysis where the relevant spaces are a topic of discussion. However, I shall in these cases simply refer to other literature.

This thesis forms a basis for a huge study that could be made on the subject. Not only are there topics to be investigated in regards to IGA on acoustic scattering, but also the coupling of IGA and absorbing boundary conditions. In this thesis I choose the infinite element method, but there are many other candidates which could work very well with IGA. Much of this study will be a topic of my PhD which will start this fall.

I would like to express my appreciation to the BeTSSi community for providing a good description for the models to be analyzed, alongside a well arranged setup for the results obtained. I would also like express my appreciation for the two books in [2] and [3] by Cotterell et al. and Frank Ihlenburg, respectively. The first has been my main reference to IGA, while the second has been my main reference for acoustic scattering using FEA.

I would like to thank NTNU for a perfect course of study, and in particular I want to thank

[1] Researcher at Norwegian Defence Research Establishment (FFI).
[2] Professor in Computational Mathematics at Department of Mathematical Sciences, NTNU.

the Department of Mathematical Sciences for providing an environment with hardware capable of computationally expensive simulations, which have been essential for this thesis. I would also thank Ilkka Karasalo and Martin Østberg[3] for providing validating data sets for the final simulations and enthusiastic support. Moreover, I would like to express my gratitude to my two supervisors Trond Jenserud and Trond Kvamsdal for their encouragement and guidance. Finally, I would like to thank Kjetil Andre Johannessen[4] for his inspirational encouragement which has not only been essential for my master thesis, but also ignited my interest in FEM/IGA in the early stages.

Jon Vegard Venås
NTNU, Trondheim
June 1, 2015

[3] Researchers at Swedish Defence Research Agency (FOI).
[4] Postdoctoral at NTNU.

Chapter 1

Introduction

1.1 Background

Acoustic scattering is a large field which has one of its application in the analysis of scattering on submarines. The scattering problem is by no means limited to submarines, as the physical phenomena occurs all around in nature. For instance, acoustic scattering may be used to calculate the number of fish in a fish farming net [4]. Moreover, the fluid to be analyzed is not limited to be water. For acoustic scattering problems, the Helmholtz equation represents the governing equation for the fluid medium. The same equation in vector form can govern electromagnetic waves ([5]). This will imply that much of the work in this thesis can be used for the study of electromagnetic waves as well.

It is fair to say that the methods available for acoustic scattering (including FEM) are not satisfactory for computation of large scale problems (due to the lack of computational efficiency). This thesis thus proposes the following problem description:

> *Investigate the contribution of IGA on acoustic scattering, and compare the results with existing methods; in particular FEA.*

The physical problem is given by the title, namely acoustic scattering. That is, sound

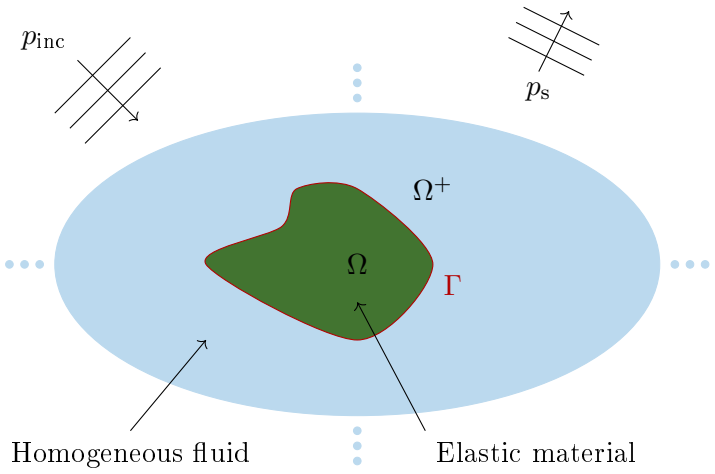


Figure 1.1: Illustration of the physical problem.

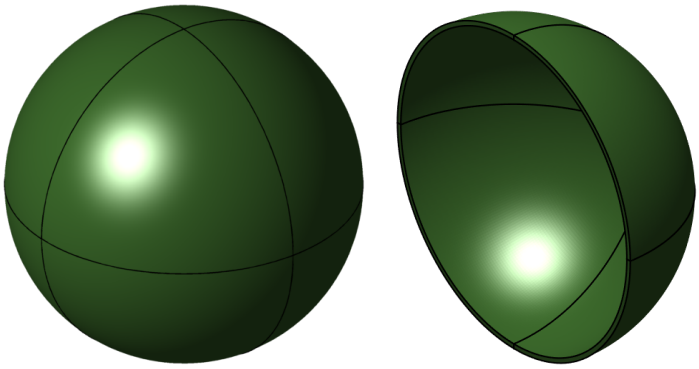


Figure 1.2: Exact geometry of a spherical shell using 8 elements.

waves in some fluid is scattered by some object. In our case, the object is made of an elastic material. The problem is illustrated in Figure 1.1 where the incoming sound waves, p_{inc}, originates from a point source far from this object such that the (spherical) sound waves are quite accurate approximated by plane waves when the waves reaches the proximity of the object. Waves scattered from the object are superimposed on the incident plane waves, which potentially could hit the object once again. As pressure waves in the fluid actuates elastic waves in the solid and vice versa, we get a fluid-structure interaction problem. We shall assume the incoming wave to be periodic in time such that the system will reach a steady-state solution. The goal is then to calculate the scattered wave p_{s} at an arbitrary far field point. Finally, to use FEM/IGA the domain must be finite, which it is not in this problem. A fictitious boundary is thus needed, which must be implemented in such a way that outgoing waves reaching this boundary are absorbed.

The geometry of the elastic object may be quite complex, but is typically exactly represented using Non-Uniform Rational B-Splines (NURBS). This fact, is one of the motivation of using IGA as it uses the same functions as basis function for analysis. The spherical shell depicted in Figure 1.2 is an example of a geometry which has an exact representation using NURBS, but is outside the space of standard finite element geometries. Moreover, IGA has proven to be superior to the classical FEA in the study of spectrum analysis (the study of vibrations).

A natural application for the acoustic scattering on submarines is the field of shape optimization. It will be illustrated that an increase in the frequency will require an corresponding increase in the computational time for FEM/IGA. So as optimization requires each iteration to be computed in a relatively small amount of time, it will restrict the optimization application to the lower frequencies. It was shown in [6] that IGA is suited for this analysis.

It has been shown that continuity of the basis functions plays an important role for elliptical problems (for instance the Helmholtz equation), see [7] and [8]. This motivates the use of IGA even further, as IGA enables the control of the continuity of the basis function, even over element boundaries (in contrast with the C^0-continuity restriction in FEA). IGA has proven to be promising in a host of areas related to the problem at hand, which yields further motivation in the use of IGA. For instance, in [9] the method was shown to be suited for the more complex scenario of sound propagation through laminar flow.

In addition to IGA the so-called infinite element method has been chosen to handle

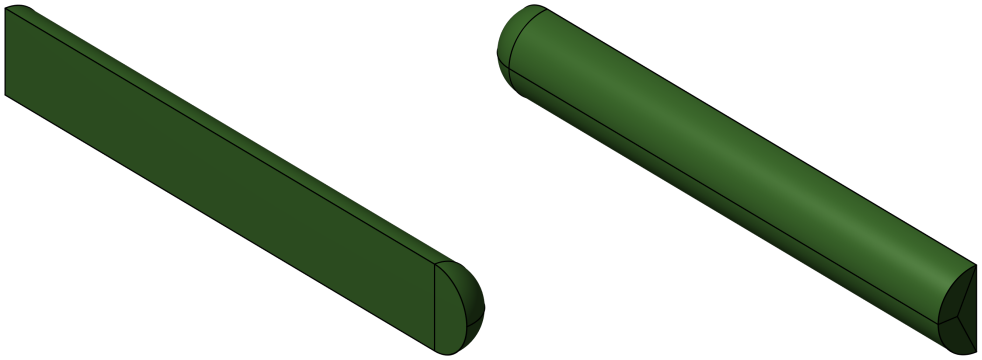


Figure 1.3: Exact geometry of BeTSSi model 1 using 9 elements.

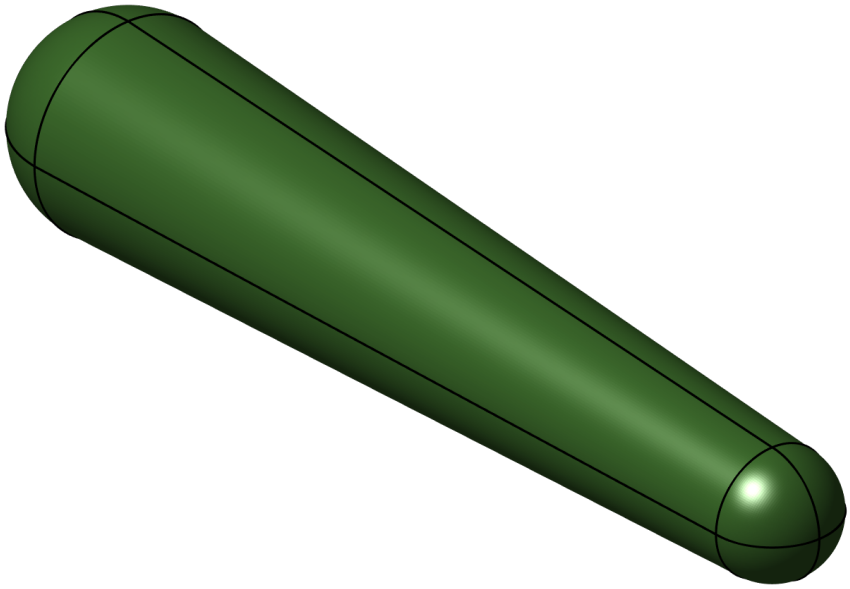


Figure 1.4: Exact geometry of BeTSSi model 3 using 12 elements.

the boundary conditions at the artificial boundary. Typically the boundary element method (BEM) has been used for this purpose with very good results. However, for larger frequencies, BEM becomes computationally expensive. Both Burnett in [10] and Gerdes and Demkowicz in [11] reports that the infinite element method is superior to BEM with respect to computational time for the same accuracy.

The Benchmark Target Strength Simulation (BeTSSi) is a workshop where several generic submarine models are analyzed by participating companies. Two of these simpler models are depicted in Figure 1.3 and Figure 1.4. The second of these workshops (description in [12]) was held in Kiel (in Germany) the fall 2014.

One of the BeTSSi models will be analyzed, namely BeTSSi model 3 depicted in Figure 1.4. We shall here limit ourselves to analysis with hard walled boundary conditions (HWBC).

A way more complex geometry is depicted in Figure 1.5 and illustrates the submarine model in the BeTSSi workshop. When dealing with such complex geometries using NURBS, one needs to introduce so called patches which glue together several NURBS objects.

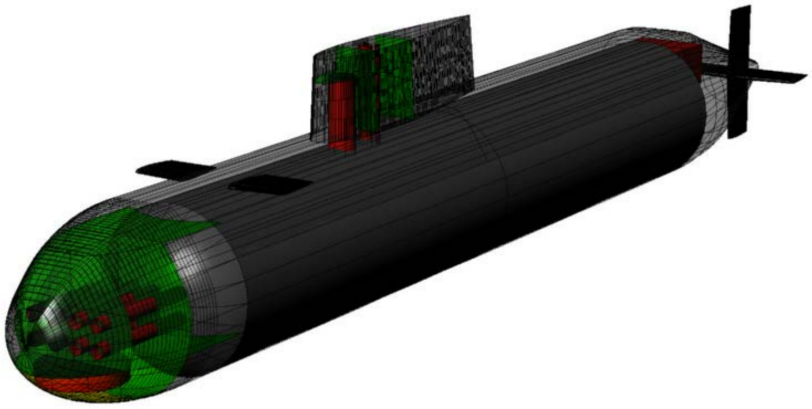


Figure 1.5: The BeTSSi submarine geometry may be exactly represented by NURBS.

One of the main goals of this thesis is to consider the so-called target strength (TS) of an object. This is a measure of the backscattered pressure from an object responding to an incident pressure field.

1.2 Outline of the thesis

We shall start by presenting the governing physical equations in Chapter 2. This is advantageus not only to give a precise mathematical definition of the equations involved, but also to present the notation used in the thesis.

Using Kirchhoff–Love shell theory it is possible to create analytic solution to the acoustic scattering problem on the spherical shell. Such analytic solution is essential to verify the developed method, and was used to verify the FEM method implemented in [3]. This thesis goes a step further in presenting the full 3D analytic solution for the elastic scattering problem of this spherical shell (Chapter 3). We are then able to compute the analytic eigenvalues for the spherical shell and the exact back scattered pressure. Moreover, we present some motivation for using the full 3D solution over the shell theory solution. The difference between the two solutions is namely non trivial.

We shall then continue in Chapter 4 by setting the stage for IGA. We start by presenting the basis functions involved, and then continue to illustrate IGA on elasticity problems.

A natural intermediate step between elastostatic analysis and elastodynamics analysis is the analysis of vibrations. In addition to illustrate the mentioned advantage IGA poses in this area, we shall analyze convergence of the eigenvalues computed by IGA for the spherical shell. This is the topic of Chapter 5.

Before starting on the full structure-fluid interaction (FSI) problem, it is important to establish good results for the infinite element method. This method only applies for the fluid, and it would thus be natural to investigate the scattering problem on rigid objects first (that is, no structure-fluid interaction occurs). An introduction to the infinite element method alongside results for the scattering on rigid sphere will be presented in Chapter 6.

With the infinite element method in our arsenal, we are now ready to consider the structure-fluid interaction problem on unbounded domains. In Chapter 7, an extensive analysis will be done for the spherical shell which will lay down a basis of understanding of the convergence analysis.

The most time consuming part of the thesis is without any doubt the programming part. With approximately 80 scripts and subroutines, and 150 functions, some 10,000 source lines of code have been implemented. As a full description of the implementation would be far to comprehensive for a thesis, only a summary of the important parts of the full fluid-structure interaction program will be described. This is the topic of Chapter 8.

In Chapter 9 we present and discuss the results obtained for BeTSSi model 3 and the full FSI problem on the spherical shell.

Finally in Chapter 10 we present the conclusions from the present work.

It must be noted that a huge amount of additional work has been moved to the appendix part for the brevity of this thesis. This includes development of several analytic solutions which has been crucial for the development of the implementations in MATLAB.

Chapter 2

Governing physical equations

The problem at hand is obviously time dependent. But we shall assume harmonic time dependency, such that all scalar function $\hat{F} = \hat{F}(\boldsymbol{x}, t)$ may be written

$$\hat{F}(\boldsymbol{x}, t) = F(\boldsymbol{x})\mathrm{e}^{-\mathrm{i}\omega t} \tag{2.1}$$

and corresponding for vector fields $\hat{\boldsymbol{F}} = \hat{\boldsymbol{F}}(\boldsymbol{x}, t)$

$$\hat{\boldsymbol{F}}(\boldsymbol{x}, t) = \boldsymbol{F}(\boldsymbol{x})\mathrm{e}^{-\mathrm{i}\omega t}$$

where ω is the angular frequency. We refer to this assumption as the *harmonic time dependency assumption*. The sign in the exponential factor is just a matter of convention. Burnett uses $\mathrm{e}^{\mathrm{i}\omega t}$ in [10], while Ihlenburg uses $\mathrm{e}^{-\mathrm{i}\omega t}$ in [3]. In this thesis, we shall use the convention after Ihlenburg.

We start by following [3, pp. 1-4] in presenting the physical equation for the problem at hand.

2.1 Acoustic waves

The fluid domain to be considered is homogeneous throughout the domain, and we shall use simplifying assumption including linearization to end up with the wave equation. From here, we move from the time domain to the frequency domain when using the harmonic time dependency assumption.

2.1.1 The continuity equation

Let $\mathcal{V}$ be a volume element in the fluid domain with boundary $\partial\mathcal{V}$ and outward normal vector $\boldsymbol{n}(\boldsymbol{x})$ where $\boldsymbol{x} \in \partial\mathcal{V}$. Denote by $\rho(\boldsymbol{x}, t)$ and $\boldsymbol{V}(\boldsymbol{x}, t)$, the mass density of the fluid in $\mathcal{V}$ and the velocity field of the fluid, respectively. Then, $\boldsymbol{V} \cdot \boldsymbol{n}$ denotes the velocity of the normal flux out of $\mathcal{V}$ through the boundary $\partial\mathcal{V}$. Conservation of mass in $\mathcal{V}$ then yields

$$\frac{\partial}{\partial t}\int_{\mathcal{V}} \rho \,\mathrm{d}\mathcal{V} = -\oint \rho\,(\boldsymbol{V}\cdot\boldsymbol{n})\,\mathrm{d}S. \tag{2.2}$$

Gauss theorem in Equation (A.1) let us rewrite the right hand side as

$$\oint \rho\,(\boldsymbol{V}\cdot\boldsymbol{n})\,\mathrm{d}S = \oint (\rho\boldsymbol{V})\cdot\boldsymbol{n}\,\mathrm{d}S = \int_{\mathcal{V}} \nabla\cdot(\rho\boldsymbol{V})\,\mathrm{d}\mathcal{V}$$

such that Equation (2.2) may be written as

$$\int_{\mathcal{V}} \frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\boldsymbol{V})\,\mathrm{d}\mathcal{V} = 0.$$

As conservation of mass is supposed to hold for any (smooth enough) domain $\mathcal{V}$ it means it has to be valid pointwise and we thus end up with the *continuity equation*

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\boldsymbol{V}) = 0. \tag{2.3}$$

2.1.2 Equation of motion

Newton's second law of motion implies that we have conservation of momentum. Thus, any change of momentum in the control volume $\mathcal{V}$ must either be due to the momentum leaving or entering the control volume, or the acts of external surface/volume forces acting on the control volume. Mathematically, this may be written as

$$\frac{\partial}{\partial t}\int_{\mathcal{V}} \rho\boldsymbol{V}\,\mathrm{d}\mathcal{V} = -\oint_{\partial\mathcal{V}} (\rho\boldsymbol{V})(\boldsymbol{V}\cdot\boldsymbol{n})\,\mathrm{d}S - \oint_{\partial\mathcal{V}} P(\boldsymbol{x},t)\boldsymbol{n}\,\mathrm{d}S + \int_{\mathcal{V}} \rho\boldsymbol{g}\,\mathrm{d}\mathcal{V} \tag{2.4}$$

were we have neglected viscous forces (note that this is a vector equation). The i^{th} component of the first term on the right hand side may (using Gauss theorem in Equation (A.1)) be written as

$$\oint_{\partial\mathcal{V}} (\rho V_i)\boldsymbol{V}\cdot\boldsymbol{n}\,\mathrm{d}\mathcal{V} = \oint_{\partial\mathcal{V}} (\rho V_i\boldsymbol{V})\cdot\boldsymbol{n}\,\mathrm{d}\mathcal{V} = \int_{\mathcal{V}} \nabla\cdot(\rho V_i\boldsymbol{V})\,\mathrm{d}\mathcal{V}$$

where V_i denotes the i^{th} component of $\boldsymbol{V}$. Using Equation (A.2), we have

$$\oint_{\partial\mathcal{V}} P(\boldsymbol{x},t)\boldsymbol{n}\,\mathrm{d}S = \int_{\mathcal{V}} \nabla P(\boldsymbol{x},t)\,\mathrm{d}\mathcal{V}.$$

Combining these surface integral transformations we may write the i^{th} component of Equation (2.4) as

$$\int_{\mathcal{V}} \frac{\partial(\rho V_i)}{\partial t} - \nabla\cdot(\rho V_i\boldsymbol{V}) + \nabla_i P(\boldsymbol{x},t) - \rho\boldsymbol{g}_i\,\mathrm{d}\mathcal{V} = 0.$$

Using the same argument as before we get

$$\frac{\partial(\rho V_i)}{\partial t} - \nabla\cdot(\rho V_i\boldsymbol{V}) = -\nabla_i P(\boldsymbol{x},t) + \rho\boldsymbol{g}_i.$$

Expanding left hand side yields

$$\rho\frac{\partial V_i}{\partial t} + \frac{\partial\rho}{\partial t}V_i + \nabla\cdot(\rho\boldsymbol{V})V_i + (\rho\boldsymbol{V})\cdot\nabla V_i = -\nabla_i P + \rho\boldsymbol{g}_i.$$

such that we may use Equation (2.2) to eliminate the second and third term of left hand side

$$\rho\frac{\partial V_i}{\partial t} + (\rho\boldsymbol{V})\cdot\nabla V_i = -\nabla_i P + \rho\boldsymbol{g}_i.$$

Combining all components we arrive at *Euler's equation*

$$\rho\frac{\partial\boldsymbol{V}}{\partial t} + \rho(\boldsymbol{V}\cdot\nabla)\boldsymbol{V} = -\nabla P + \rho\boldsymbol{g}. \tag{2.5}$$

2.1.3 Helmholtz equation

We first need to linearize the continuity equation in Equation (2.3) and Euler's equation in Equation (2.5). This is done by setting

$$P = P_0 + P_1, \quad \rho = \rho_0 + \rho_1, \quad \text{and} \quad \boldsymbol{V} = \boldsymbol{V}_0 + \boldsymbol{V}_1 \tag{2.6}$$

where parameters with subscript 0 is equilibrium parameters and parameters with subscript 1 refer to small perturbed parameters. That is, we assume there is only a small perturbation of a background field for the pressure, density and the velocity field. Assuming there is no background velocity, we have $\boldsymbol{V}_0 = \boldsymbol{0}$. We shall also assume the background density to be constant, such that

$$\frac{\partial \rho_0}{\partial t} = 0 \quad \text{and} \quad \nabla \rho_0 = \boldsymbol{0}.$$

The background pressure field can only be due to the hydrostatic pressure from the gravity force density ($\rho \boldsymbol{g} = -\rho g \boldsymbol{e}_3$) such that we have

$$P_0 = P_\text{a} - \rho z g.$$

where P_a is the (constant) background pressure at $z = 0$. This implies that

$$\frac{\partial P_0}{\partial t} = 0 \quad \text{and} \quad \nabla P_0 = \rho \boldsymbol{g}.$$

Euler's equation thus simplifies to

$$\rho \frac{\partial \boldsymbol{V}}{\partial t} + \rho(\boldsymbol{V} \cdot \nabla)\boldsymbol{V} = -\nabla P_1. \tag{2.7}$$

The linearization procedure is then completed by inserting the expression in Equation (2.6), and neglect the terms of second order or higher. For the continuity equation in Equation (2.3), we have

$$\frac{\partial \rho_1}{\partial t} + \nabla \cdot ((\rho_0 + \rho_1)\boldsymbol{V}_1) = 0$$

which after linearization yields the *linearized version* of the continuity equation

$$\frac{\partial \rho_1}{\partial t} + \rho_0 \nabla \cdot \boldsymbol{V}_1 = 0. \tag{2.8}$$

For the simplified Euler equation in Equation (2.7) we get

$$(\rho_0 + \rho_1)\frac{\partial \boldsymbol{V}_1}{\partial t} + \rho(\boldsymbol{V}_1 \cdot \nabla)\boldsymbol{V}_1 = -\nabla P_1$$

which after linearization yields the *linearized version* of the Euler equation

$$\rho_0 \frac{\partial \boldsymbol{V}_1}{\partial t} = -\nabla P_1. \tag{2.9}$$

We shall assume a linear material law,

$$P = c^2 \rho \qquad \Rightarrow \qquad \frac{\partial P_1}{\partial t} = c^2 \frac{\partial \rho_1}{\partial t},$$

where c is the speed of sound (material parameter which we assume to be constant). Using Equation (2.8) and Equation (2.9), we get

$$\begin{aligned}\frac{\partial^2 P_1}{\partial t^2} &= c^2 \frac{\partial^2 \rho_1}{\partial t^2} = c^2 \frac{\partial}{\partial t}\left(\frac{\partial \rho_1}{\partial t}\right) = c^2 \frac{\partial}{\partial t}\left(-\rho_0 \nabla \cdot \boldsymbol{V}_1\right) = -c^2 \nabla \cdot \left(\rho_0 \frac{\partial \boldsymbol{V}_1}{\partial t}\right) \\ &= -c^2 \nabla \cdot (-\nabla P_1) = c^2 \Delta P_1\end{aligned}$$

which is simply the classical wave equation for the pressure

$$\frac{1}{c^2}\frac{\partial^2 P_1}{\partial t^2} = \Delta P_1. \tag{2.10}$$

Inserting $P_1(\boldsymbol{x}, t) = \bar{P}_1(\boldsymbol{x}, k)\mathrm{e}^{-\mathrm{i}\omega t}$ then yields the *Helmholtz equation*

$$\Delta \hat{P}_1 + k^2 \hat{P}_1 = 0$$

where $k = \frac{\omega}{c}$ is called the *wave number*.

2.2 Linear elasticity

One important assumption for using linear elasticity is that only small deformations of the material occurs. We will here not show in detail the derivation of the governing equation for linearized elasticity, but we present the notation used in the formulation of the isogeometric analysis.

The notations used, takes inspirations from [2]. In this section, the indices i, j, k and l will denote a specific spatial direction. All calculations will be in three dimensions, such that $i, j, k, l = 1, 2, 3$. Moreover, u_i shall denote the i^{th} component of the vector $\boldsymbol{u}$ and differentiation is denoted with a comma such that

$$u_{i,j} = u_{i,x_j} = \frac{\partial u_i}{\partial x_j}.$$

Finally, we use the convention that if an index is repeated, it imply summation. That is,

$$\sigma_{ij} n_j = \sigma_{i,1} n_1 + \sigma_{i,2} n_2 + \sigma_{i,3} n_3$$

and

$$\sigma_{ij,j} + f_i = \frac{\partial \sigma_{i1}}{\partial x_1} + \frac{\partial \sigma_{i2}}{\partial x_2} + \frac{\partial \sigma_{i3}}{\partial x_3} + f_i.$$

Note that we do not sum over i in the latter example since the quantities are separated by a plus sign. Define now the *symmetric part* of a general tensor $\boldsymbol{A} = [A_{ij}]$ to be

$$A_{(ij)} = A_{(ji)} := \frac{A_{ij} + A_{ji}}{2}$$

and note that if $\boldsymbol{B} = [B_{ij}] = [B_{(ij)}]$ is a symmetric tensor, then

$$A_{ij} B_{ij} = A_{(ij)} B_{ij}. \tag{2.11}$$

That is, we can combine the components of $\boldsymbol{B}$ which are equal to reduce redundant computations.

Let now σ_{ij} denote the Cartesian components of the Cauchy *stress tensor* and let ε_{ij} denote the *infinitesimal strain tensor* which is defined by

$$\varepsilon_{ij} = u_{(i,j)} = \frac{u_{i,j} + u_{j,i}}{2}.$$

We can now state the relation between ε_{ij} as σ_{ij} using the generalized Hooke's law as

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl}$$

where c_{ijkl} are *elastic coefficients*. In the case of isotropic material, these coefficients are given by

$$c_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)$$

where the Kronecker delta function is given by

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & \text{otherwise} \end{cases}$$

and the parameters λ and μ are the *Lamé parameters* which are expressed in terms of E and ν (*Young's modulus* and *Poisson's ratio* respectively) by

$$\lambda = \frac{\nu E}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)}.$$

We are now ready to state the *strong form* of the linear elasticity problem in three dimensions.

Let $\Omega \subset \mathbb{R}^3$ be the domain with a boundary $\partial\Omega$ which is composed of two parts; Γ_{D_i} and Γ_{N_i}. These are called *Dirichlet* and *Neumann* boundary conditions, respectively, and satisfies $\bigcup_{i=1}^{3} \Gamma_{D_i} \cup \Gamma_{N_i} = \partial\Omega$ and $\Gamma_{D_i} \cap \Gamma_{N_i} = \emptyset$. Moreover, let the functions $f_i : \Omega \to \mathbb{R}$, $g_i : \Gamma_{D_i} \to \mathbb{R}$ and $h_i : \Gamma_{N_i} \to \mathbb{R}$ be given. Then, find $u_i : \overline{\Omega} \to \mathbb{R}$ such that

$$\sigma_{ij,j} + f_i = \rho_s u_{i,tt} \qquad \text{in} \quad \Omega, \tag{2.12}$$

$$u_i = g_i \qquad \text{on} \quad \Gamma_{D_i}, \tag{2.13}$$

$$\sigma_{ij} n_j = h_i \qquad \text{on} \quad \Gamma_{N_i}, \tag{2.14}$$

for $i = 1, 2, 3$.

Using the assumption of periodicity, we may insert $u_i \to u_i \mathrm{e}^{-\mathrm{i}\omega t}$ (and $f_i \to f_i \mathrm{e}^{-\mathrm{i}\omega t}$) into Equation (2.12). As σ is a linear function of u_i, we get a common factor $\mathrm{e}^{-\mathrm{i}\omega t}$ which simplifies the equation to

$$\sigma_{ij,j} + \omega^2 \rho_s u_i = -f_i \qquad \text{in} \quad \Omega. \tag{2.15}$$

Chapter 3

Analytic solution for spherical shell

There are few known analytic solution to full 3D scattering problems, but for the spherical shell, the exact solution is known (the exact solution is also known for a solid sphere, but this case will not be considered). Whenever the spherical shell is analyzed, we shall mainly use the parameters found in Table 3.1. It is typically also possible to construct analytic solution to an in homogeneous Helmholtz equation by inserting an arbitrary smooth solution and calculate the resulting right hand side. Correspondingly for elasticity problems with body forces. However, for the homogeneous equations one does not have this liberty. Elastic scattering on a spherical shell thus serves as a valuable test problem before geometries with higher complexity may be analyzed.

3.1 Fundamental functions

Exact solutions for the spherical shell are heavily based on the spherical coordinate system defined in Appendix B.3. Some fundamental functions then naturally arise, and we shall briefly present their notation in the following.

As the *Bessel functions* (or *cylindrical harmonics*) are implemented in MATLAB, we shall use these functions as a starting point.

Table 3.1: Parameters for spherical shell.

Parameter	Description
$E = 2.07 \cdot 10^{11}\text{Pa}$	Young's modulus
$\rho_s = 7669\text{kg/m}^3$	Density of solid
$\rho_\text{f} = 1000\text{kg/m}^3$	Density of fluid
$\nu = 0.3$	Poisson's ratio
$c_\text{f} = 1524\text{m/s}$	Fluid speed of sound
$t = 0.15\text{m}$	Thickness of the shell
$R = 5\text{m}$	Radius of the midsurface of the shell
$R_\text{o} = 5.075\text{m}$	Outer radius of the shell
$P_0 = 1$	Amplitude of incident wave

3.1.1 Legendre polynomials

The Legendre polynomials are defined recursively by (cf. [13, p. 332])

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

starting with $P_0(x) = 1$ and $P_1(x) = x$. The first five of these two sets of functions are illustrated in Figure 3.1. The solutions presented involving these functions appears as $P_n(\cos\theta)$, so we also plot these functions in Figure 3.2.

3.1.2 Bessel functions

The Bessel functions of the first kind are defined by

$$J_\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m!\Gamma(m+\alpha+1)} \left(\frac{x}{2}\right)^{2m+\alpha},$$

while the Bessel functions of the second kind are defined by

$$Y_\alpha(x) = \frac{J_\alpha \cos(\alpha\pi) - J_{-\alpha}(x)}{\sin(\alpha\pi)},$$

where

$$Y_n(x) = \lim_{\alpha \to n} Y_\alpha(x)$$

whenever $n \in \mathbb{Z}$ (cf. [13, p. 358]). The derivatives of these functions are given by

$$J'_\alpha(x) = \begin{cases} \frac{1}{2}\left(J_{\alpha-1} - J_{\alpha+1}\right) & \alpha \neq 0 \\ -J_1(x) & \alpha = 0 \end{cases}$$

and

$$Y'_\alpha(x) = \begin{cases} \frac{1}{2}\left(Y_{\alpha-1} - Y_{\alpha+1}\right) & \alpha \neq 0 \\ -Y_1(x) & \alpha = 0. \end{cases}$$

The first five functions of these two sets of functions are illustrated in Figure 3.3 and Figure 3.4, respectively.

We may now use these definitions to define the *spherical Bessel functions*. The spherical Bessel functions of the first kind are defined by (cf. [13, p. 437])

$$j_n(x) = \sqrt{\frac{\pi}{2x}} J_{n+\frac{1}{2}}(x)$$

and the second kind are defined by

$$y_n(x) = \sqrt{\frac{\pi}{2x}} Y_{n+\frac{1}{2}}(x).$$

The first five functions of these two sets of functions are illustrated in Figure 3.5 and Figure 3.6. The derivatives of these functions are simply found by the product rule

$$j'_n(x) = \sqrt{\frac{\pi}{2x}} J'_{n+\frac{1}{2}}(x) - \frac{1}{2}\sqrt{\frac{\pi}{2}} \frac{1}{x^{3/2}} J_{n+\frac{1}{2}}(x)$$

and

$$y'_n(x) = \sqrt{\frac{\pi}{2x}} Y'_{n+\frac{1}{2}}(x) - \frac{1}{2}\sqrt{\frac{\pi}{2}} \frac{1}{x^{3/2}} Y_{n+\frac{1}{2}}(x).$$

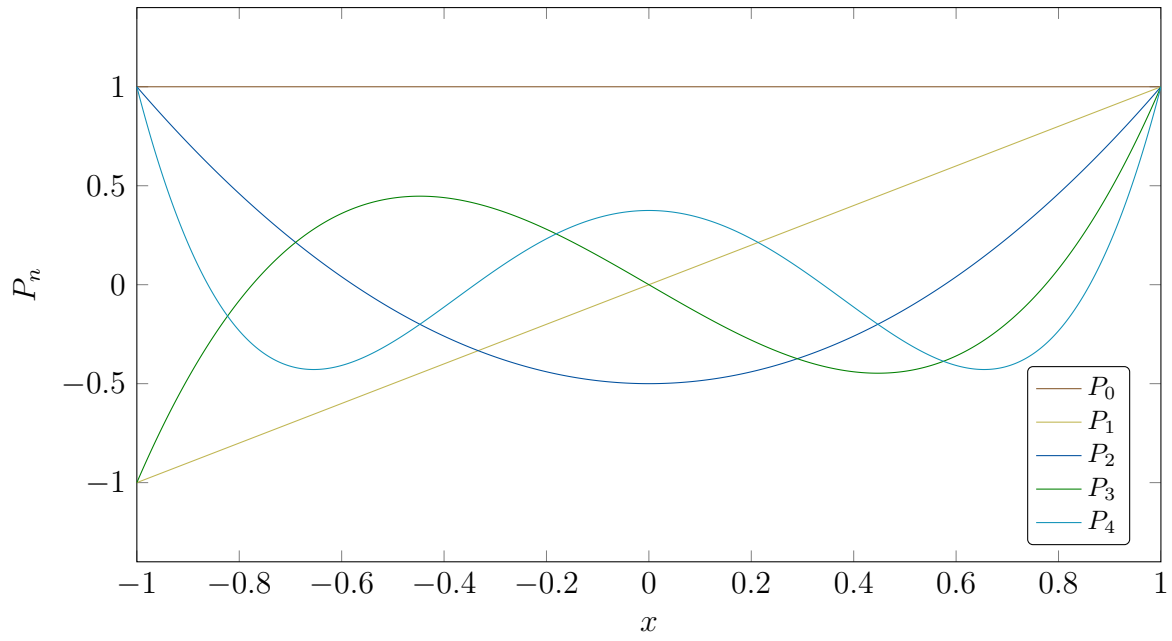


Figure 3.1: Legendre polynomials: The first 5 Legendre polynomials $P_n(x)$.

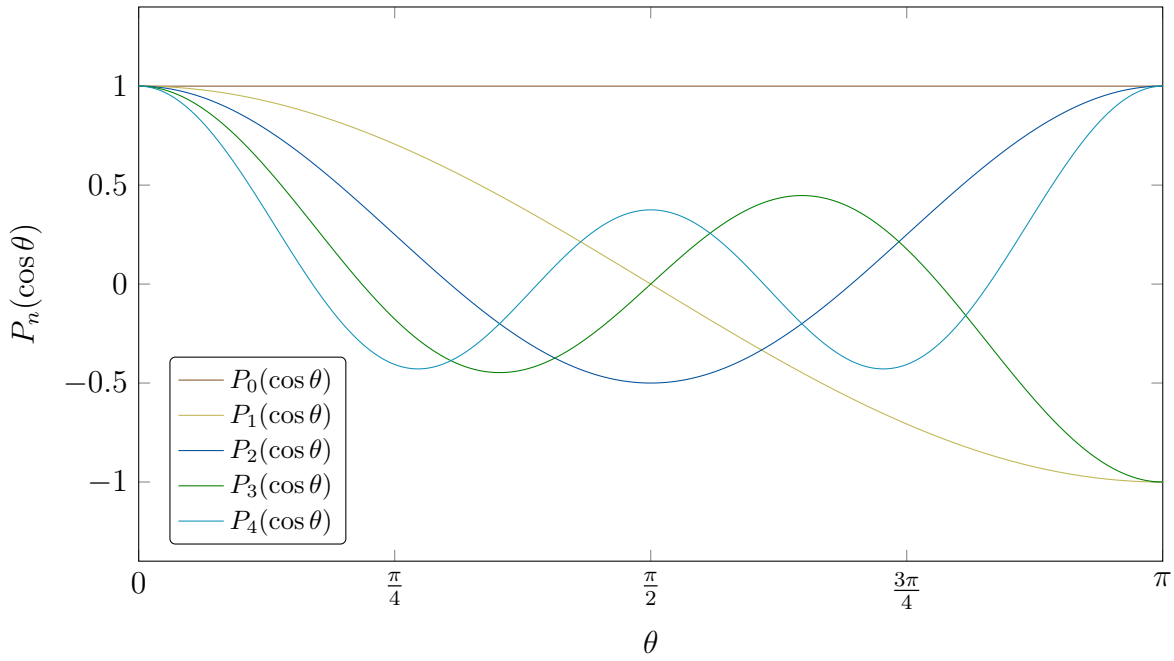


Figure 3.2: Legendre polynomials: The first 5 Legendre polynomials P_n evaluated at $\cos\theta$ for $\theta \in [0, \pi]$.

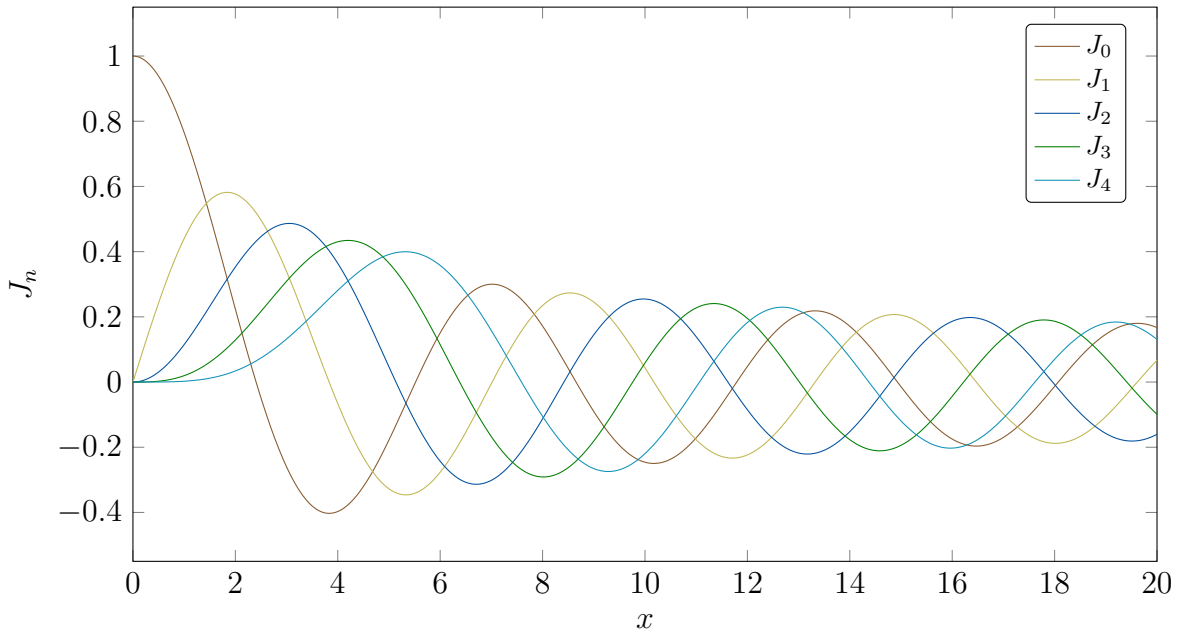


Figure 3.3: Bessel function of first kind: The first 5 Bessel functions of first kind $J_n(x)$.

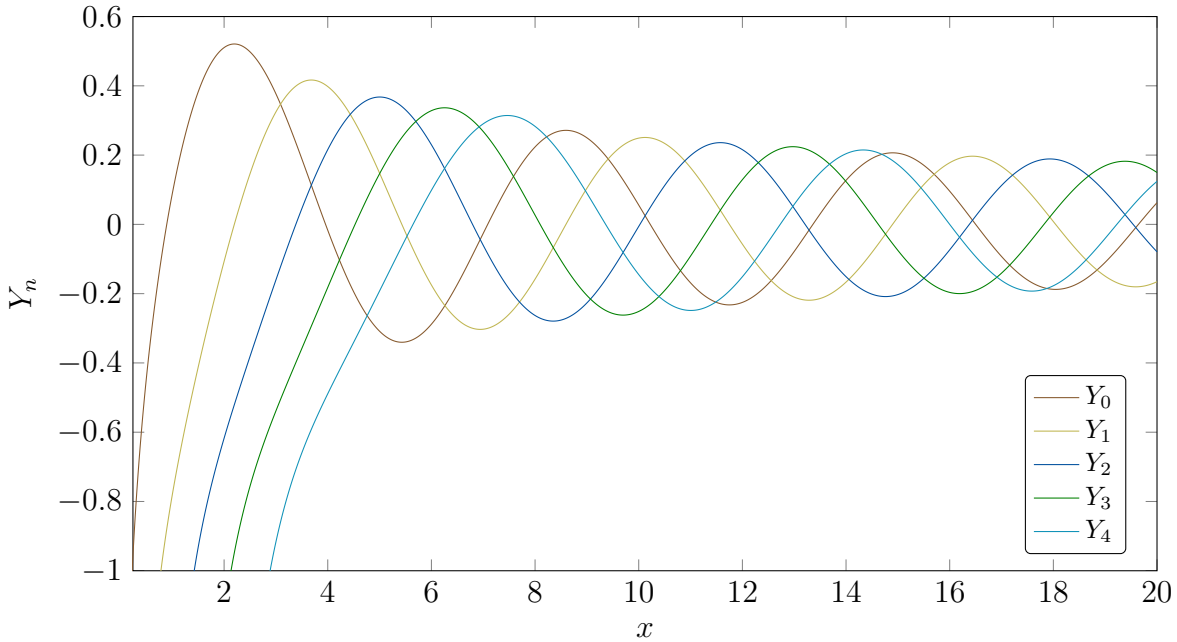


Figure 3.4: Bessel function of second kind: The first 5 Bessel functions of second kind $Y_n(x)$.

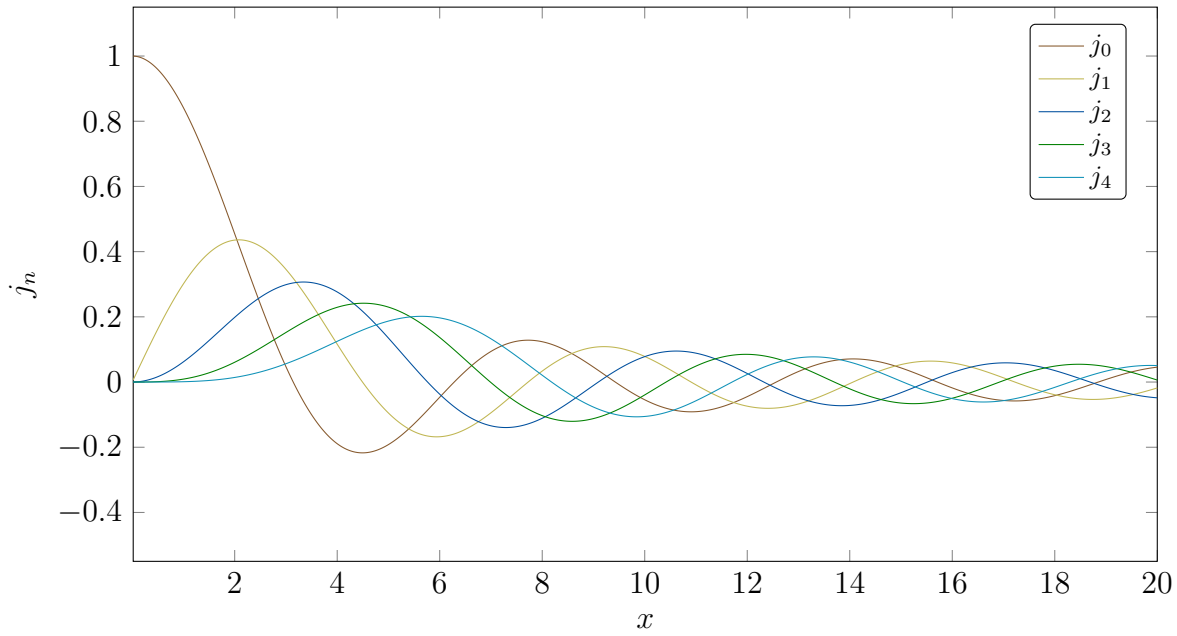


Figure 3.5: Spherical Bessel function of first kind: The first 5 Spherical Bessel functions of first kind $j_n(x)$.

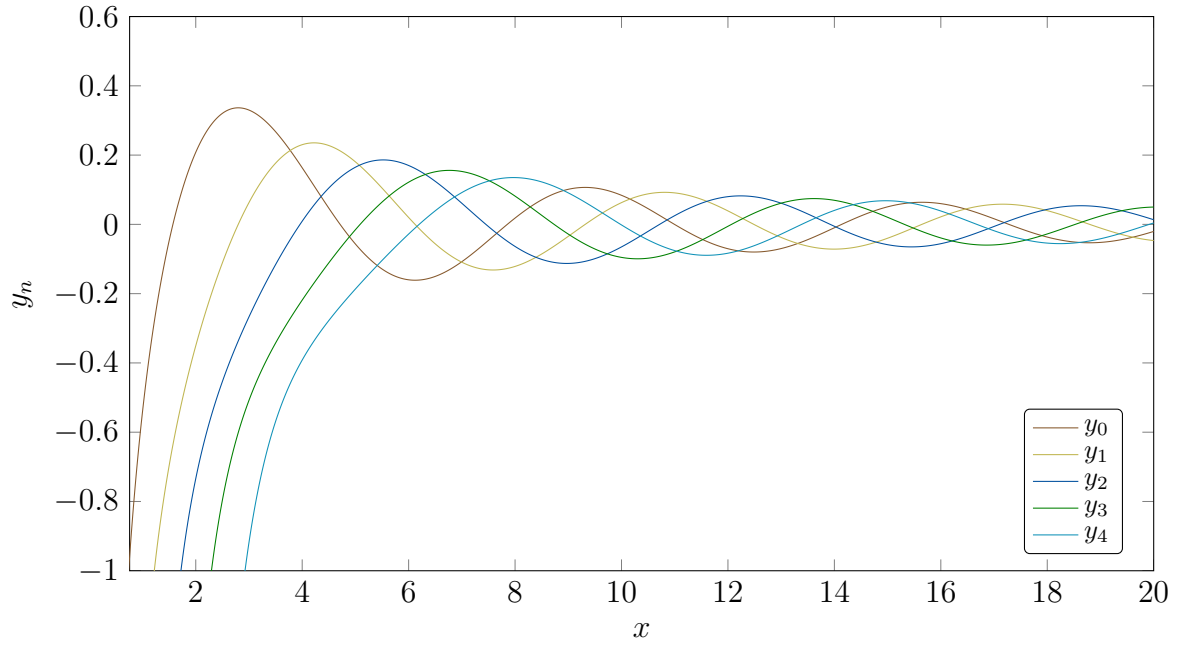


Figure 3.6: Spherical Bessel function of second kind: The first 5 Spherical Bessel function of second kind $y_n(x)$.

3.1.3 Spherical Hankel functions

The spherical Hankel functions of the first kind are now given by

$$h_n(x) = j_n(x) + \mathrm{i}y_n(x)$$

with a derivative given by

$$h_n'(x) = j_n'(x) + \mathrm{i}y_n'(x).$$

The spherical Hankel functions of the second kind are simply defined by

$$h_n^{(2)}(x) = j_n(x) - \mathrm{i}y_n(x).$$

As these solution will represent in-going waves in scattering problems, they will not be used (They will be eliminated by the Sommerfield condition, which we will come back to).

3.2 Simplifying notations

In [14, pp. 12-20] the exact 3D elasticity solution for the spherical shell is presented. We shall repeat the final formulas here, but will adopt them to the notation of the thesis and do some simplifications in the expressions. As the formulas involved are quite comprehensive, we shall start by defining some simplifying notation. For convenience, we define

$$Z_n^{(1)}(x) = j_n(x), \quad \text{and} \quad Z_n^{(2)}(x) = y_n(x),$$

such that we can define

$$\begin{aligned}
U_1^{(i)}(x) &= nZ_n^{(i)}(x) - xZ_{n+1}^{(i)}(x) \\
U_3^{(i)}(x) &= n(n+1)Z_n^{(i)}(x) \\
T_{11}^{(i)}(x) &= \left[n^2 - n - \frac{1}{2}\left(\frac{\beta}{\alpha}\right)^2 x^2\right] Z_n^{(i)}(x) + 2xZ_{n+1}^{(i)}(x) \\
T_{13}^{(i)}(x) &= n(n+1)\left[(n-1)Z_n^{(i)}(x) - xZ_{n+1}^{(i)}(x)\right] \\
T_{21}^{(i)}(x) &= \left[-n^2 - \frac{1}{2}\left(\frac{\beta}{\alpha}\right)^2 x^2 + x^2\right] Z_n^{(i)}(x) - xZ_{n+1}^{(i)}(x) \\
T_{23}^{(i)}(x) &= -(n^2+n)\left[nZ_n^{(i)}(x) - xZ_{n+1}^{(i)}(x)\right] \\
T_{31}^{(i)}(x) &= \left[n - \frac{1}{2}x^2 + \left(\frac{\alpha}{\beta}\right)^2 x^2\right] Z_n^{(i)}\left(\frac{\alpha}{\beta}x\right) - \frac{\alpha}{\beta}xZ_{n+1}^{(i)}\left(\frac{\alpha}{\beta}x\right) \\
T_{33}^{(i)}(x) &= n(n+1)Z_n^{(i)}(x) \\
T_{41}^{(i)}(x) &= (n-1)Z_n^{(i)}(x) - xZ_{n+1}^{(i)}(x) \\
T_{43}^{(i)}(x) &= \left(n^2 - 1 - \frac{1}{2}x^2\right) Z_n^{(i)}(x) + xZ_{n+1}^{(i)}(x)
\end{aligned}$$

where

$$\alpha = \frac{\omega}{c_1}, \quad \beta = \frac{\omega}{c_2}, \quad c_1 = \sqrt{\frac{\lambda + 2\mu}{\rho_s}}, \quad c_2 = \sqrt{\frac{\mu}{\rho_s}}.$$

Moreover, we define the coefficients A_n, B_n, C_n and D_n by

$$A_0 = \frac{\frac{R_o^2}{2\mu} T_{11}^{(2)}(\alpha R_i)}{\Delta_0}, \quad B_0 = 0, \quad C_0 = -\frac{\frac{R_o^2}{2\mu} T_{11}^{(1)}(\alpha R_i)}{\Delta_0}, \quad D_0 = 0$$

for $n = 0$ where

$$\Delta_0 = \begin{vmatrix} T_{11}^{(1)}(\alpha R_o) & T_{11}^{(2)}(\alpha R_o) \\ T_{11}^{(1)}(\alpha R_i) & T_{11}^{(2)}(\alpha R_i) \end{vmatrix},$$

and for $n > 0$ we have

$$A_n = \frac{\Delta_{n1}}{\Delta_n}, \quad B_n = \frac{\Delta_{n2}}{\Delta_n}, \quad C_n = \frac{\Delta_{n3}}{\Delta_n}, \quad D_n = \frac{\Delta_{n4}}{\Delta_n}$$

where

$$\Delta_n = \begin{vmatrix} T_{11}^{(1)}(\alpha R_o) & T_{13}^{(1)}(\beta R_o) & T_{11}^{(2)}(\alpha R_o) & T_{13}^{(2)}(\beta R_o) \\ T_{11}^{(1)}(\alpha R_i) & T_{13}^{(1)}(\beta R_i) & T_{11}^{(2)}(\alpha R_i) & T_{13}^{(2)}(\beta R_i) \\ T_{41}^{(1)}(\alpha R_o) & T_{43}^{(1)}(\beta R_o) & T_{41}^{(2)}(\alpha R_o) & T_{43}^{(2)}(\beta R_o) \\ T_{41}^{(1)}(\alpha R_i) & T_{43}^{(1)}(\beta R_i) & T_{41}^{(2)}(\alpha R_i) & T_{43}^{(2)}(\beta R_i) \end{vmatrix} \tag{3.1}$$

$$\Delta_{n1} = \frac{R_o^2}{2\mu} \begin{vmatrix} T_{13}^{(1)}(\beta R_i) & T_{11}^{(2)}(\alpha R_i) & T_{13}^{(2)}(\beta R_i) \\ T_{43}^{(1)}(\beta R_o) & T_{41}^{(2)}(\alpha R_o) & T_{43}^{(2)}(\beta R_o) \\ T_{43}^{(1)}(\beta R_i) & T_{41}^{(2)}(\alpha R_i) & T_{43}^{(2)}(\beta R_i) \end{vmatrix}$$

$$\Delta_{n2} = -\frac{R_o^2}{2\mu} \begin{vmatrix} T_{11}^{(1)}(\alpha R_i) & T_{11}^{(2)}(\alpha R_i) & T_{13}^{(2)}(\beta R_i) \\ T_{41}^{(1)}(\alpha R_o) & T_{41}^{(2)}(\alpha R_o) & T_{43}^{(2)}(\beta R_o) \\ T_{41}^{(1)}(\alpha R_i) & T_{41}^{(2)}(\alpha R_i) & T_{43}^{(2)}(\beta R_i) \end{vmatrix}$$

$$\Delta_{n3} = \frac{R_o^2}{2\mu} \begin{vmatrix} T_{11}^{(1)}(\alpha R_i) & T_{13}^{(1)}(\beta R_i) & T_{13}^{(2)}(\beta R_i) \\ T_{41}^{(1)}(\alpha R_o) & T_{43}^{(1)}(\beta R_o) & T_{43}^{(2)}(\beta R_o) \\ T_{41}^{(1)}(\alpha R_i) & T_{43}^{(1)}(\beta R_i) & T_{43}^{(2)}(\beta R_i) \end{vmatrix}$$

$$\Delta_{n4} = -\frac{R_o^2}{2\mu} \begin{vmatrix} T_{11}^{(1)}(\alpha R_i) & T_{13}^{(1)}(\beta R_i) & T_{11}^{(2)}(\alpha R_i) \\ T_{41}^{(1)}(\alpha R_o) & T_{43}^{(1)}(\beta R_o) & T_{41}^{(2)}(\alpha R_o) \\ T_{41}^{(1)}(\alpha R_i) & T_{43}^{(1)}(\beta R_i) & T_{41}^{(2)}(\alpha R_i) \end{vmatrix}.$$

The *mechanical impedance* is now defined by

$$Z_n = \frac{R_o}{-i\omega \left[A_n U_1^{(1)}(\alpha R_o) + C_n U_1^{(2)}(\alpha R_o) + B_n U_3^{(1)}(\beta R_o) + D_n U_3^{(2)}(\beta R_o) \right]}$$

and the *specific acoustic impedance* is given by

$$z_n = i\rho_f c_f \frac{h_n(kR_o)}{h_n'(kR_o)}.$$

Table 3.2: Vibration of spherical shell in vacuum: Eigenvalues.

n	ω_{n1}	ω_{n2}	ω_{n3}	ω_{n4}
0	1 756.512 808	126 260.364 283	252 500.986 439	378 745.996 169
1	2 150.627 736	67 515.475 434	126 228.461 764	135 011.522 862
2	764.120 551	2 963.645 770	67 557.845 627	126 165.580 365
3	907.966 763	3 957.249 140	67 621.308 959	126 073.458 170
4	972.169 223	5 003.665 721	67 705.755 978	125 954.457 998
5	1 017.829 344	6 068.816 531	67 811.042 230	125 811.341 874
6	1 065.383 457	7 142.244 424	67 936.989 482	125 647.053 072
7	1 124.910 447	8 219.893 517	68 083.387 218	125 464.535 596
8	1 202.487 436	9 299.877 139	68 249.994 352	125 266.603 859
9	1 301.807 206	10 381.189 705	68 436.541 152	125 055.861 655
10	1 424.763 754	11 463.235 315	68 642.731 323	124 834.661 792
11	1 571.893 542	12 545.628 972	68 868.244 217	124 605.095 229
12	1 742.817 050	13 628.103 050	69 112.737 138	124 368.999 333
13	1 936.628 939	14 710.459 324	69 375.847 706	124 127.977 066
14	2 152.185 866	15 792.542 594	69 657.196 233	123 883.421 272
15	2 388.286 746	16 874.225 327	69 956.388 103	123 636.540 322
16	2 643.769 233	17 955.398 269	70 273.016 103	123 388.382 902
17	2 917.551 993	19 035.964 470	70 606.662 699	123 139.860 776
18	3 208.646 463	20 115.835 338	70 956.902 215	122 891.769 039
19	3 516.153 537	21 194.927 938	71 323.302 920	122 644.803 765
20	3 839.254 092	22 273.163 098	71 705.428 985	122 399.577 175

3.3 Vibration of spherical shell in vacuum

The eigenvalues of a spherical elastic shell in vacuum is found by solving $\Delta_n = 0$ (given by Equation (3.1)) for all *modes* n. For each mode n there will be infinitely many eigenvalues. For a given n, we denote by ω_{nm} the eigenvalue corresponding to the m^{th} zero of Δ_n. The resulting eigenvalues seems to grow as n grows for large n (as shown in Figure 3.7). In Table 3.2 we show the first 4 zeros of Δ_n for n from 0 to 20. The first four (lowest) eigenvalues are found to be $\omega_{21} = 764.12055$, $\omega_{31} = 907.96676$, $\omega_{41} = 972.16922$ and $\omega_{51} = 1017.829$.

3.4 Scattering on a rigid sphere

Let the plane wave

$$p_{\mathrm{inc}}(r,\theta) = P_0 \mathrm{e}^{\mathrm{i}kz} = P_0 \mathrm{e}^{\mathrm{i}kr\cos\theta}$$

be scattered on a sphere with radius R. The resulting scattered wave is then

$$p_{\mathrm{s}\infty}(r,\theta) = -P_0 \sum_{n=0}^{\infty} \mathrm{i}^n (2n+1) P_n(\cos\theta) \frac{j_n'(kR)}{h_n'(kR)} h_n(kr).$$

For details of this solution see [3, p. 28].

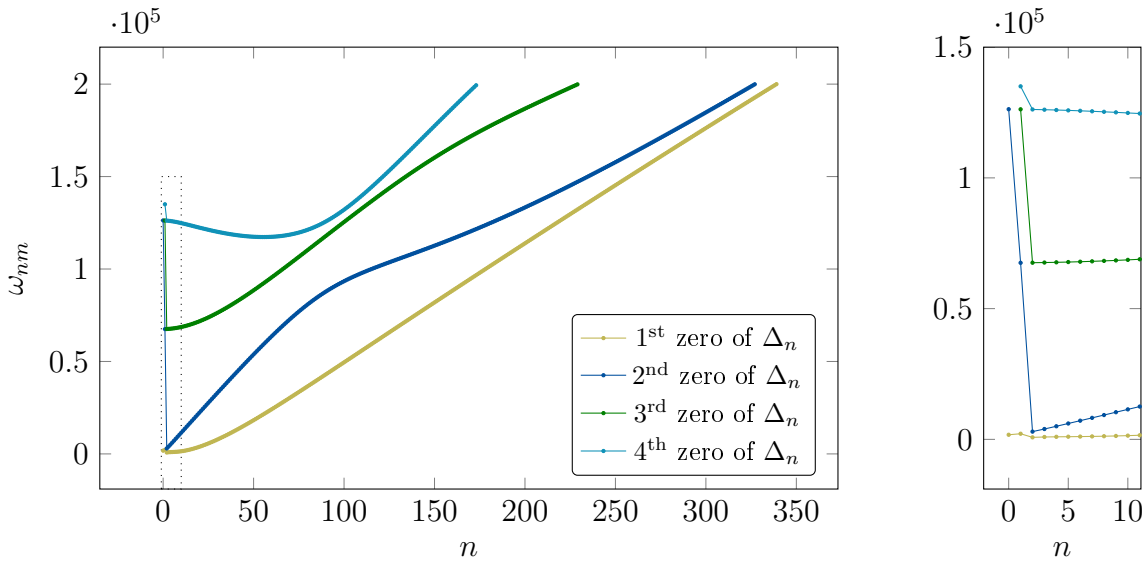


Figure 3.7: Vibrations of spherical shell in vacuum: All eigenvalues below $2 \cdot 10^5$, which results in 1186 eigenvalues. The plot on the right shows a magnification of the domain inside the dotted lines.

3.5 Scattering on elastic spherical shell

The full 3D exact solution to the elastic scattering problem is given by

$$p(r,\theta) = p_{\mathrm{s},\infty}(r,\theta) + p_{\mathrm{s,r}}(r,\theta) \tag{3.2}$$

where

$$p_{\mathrm{s},\infty}(r,\theta) = -P_0 \sum_{n=0}^{\infty} (2n+1)\mathrm{i}^n P_n(\cos\theta) \frac{j_n'(kR_\mathrm{o})}{h_n'(kR_\mathrm{o})} h_n(kr)$$

and

$$p_{\mathrm{s,r}}(r,\theta) = \frac{P_0 \rho_\mathrm{f} c_\mathrm{f}}{(kR_\mathrm{o})^2} \sum_{n=0}^{\infty} \frac{\mathrm{i}^n (2n+1) P_n(\cos\theta) h_n(kr)}{[h_n'(kR_\mathrm{o})]^2 (Z_n + z_n)}.$$

Here, $p_{\mathrm{s},\infty}$ represent a scattered wave from a rigid body, and $p_{\mathrm{s,r}}$ represent the pressure corresponding to the forced vibrations of the elastic body in fluid (loaded with the sum of incident pressure p_inc and the rigid body scattered pressure $p_{\mathrm{s},\infty}$) (see [14, p. 2] for more details). Note that we do not include the incident pressure field p_inc in p, as this field is not present in the numerical solution. However, to compute the total physical pressure at the wet surface this field must be included (in addition to the constant background pressure).

In Figure 3.9 we plot the modulus of the function

$$F(k) = r_\mathrm{f} \mathrm{e}^{\mathrm{i}kr_\mathrm{f}} p(r_\mathrm{f}, \alpha_\mathrm{f}, \beta_\mathrm{f})$$

where $\boldsymbol{P}_\mathrm{f}$ is the far field point which depends on r_f, α_f and β_f (cf. Figure 3.8). This plot illustrates the complexity of the exact solution at hand. It seems to be discontinuous at the eigenmodes (with wave numbers found in Table 3.3), but if zoomed in enough, we observe the spikes to be smooth as well. As Figure 3.9 illustrates, we must magnify a lot to show details of the higher eigenmodes. If such a detailed plot was to be illustrated using

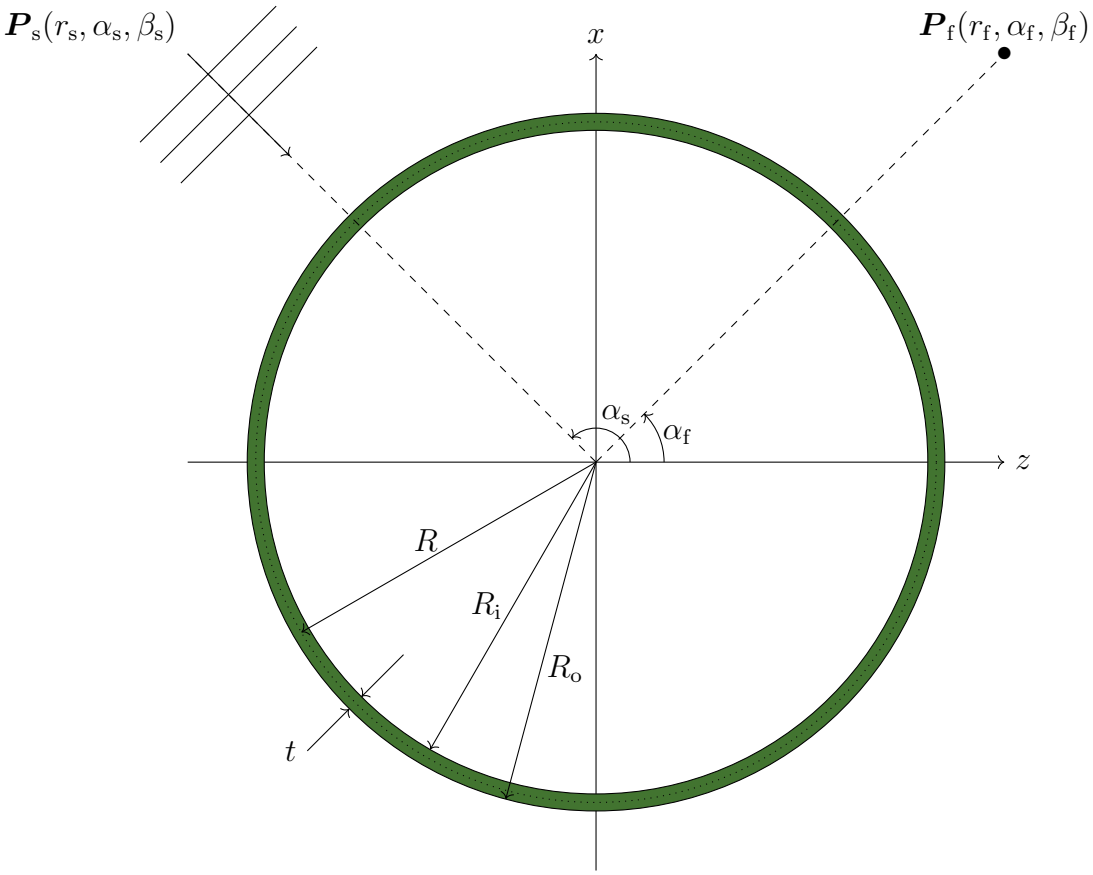


Figure 3.8: Scattering on elastic spherical shell: The spherical shell in the xz-plane. Here with $R = 5$ and $t = 0.25$ (the calculations uses $t = 0.15$ and are here set to 0.25 only for visualization purposes).

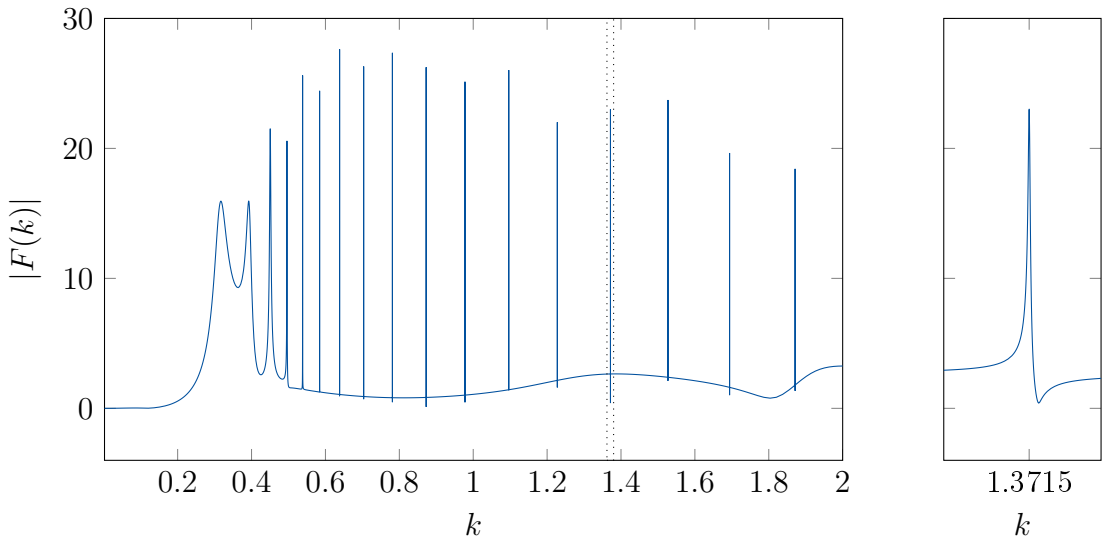


Figure 3.9: Scattering on elastic spherical shell: Far-field pattern of backscattered pressure (we set $\alpha_\mathrm{s} = \alpha_\mathrm{f} = \pi$) for elastic spherical shell (monostatic calculation). The plot on the right shows a magnification (in x-direction) of the domain inside the dotted lines. The length of the domain of this plot is $\Delta k = 10^{-6}$.

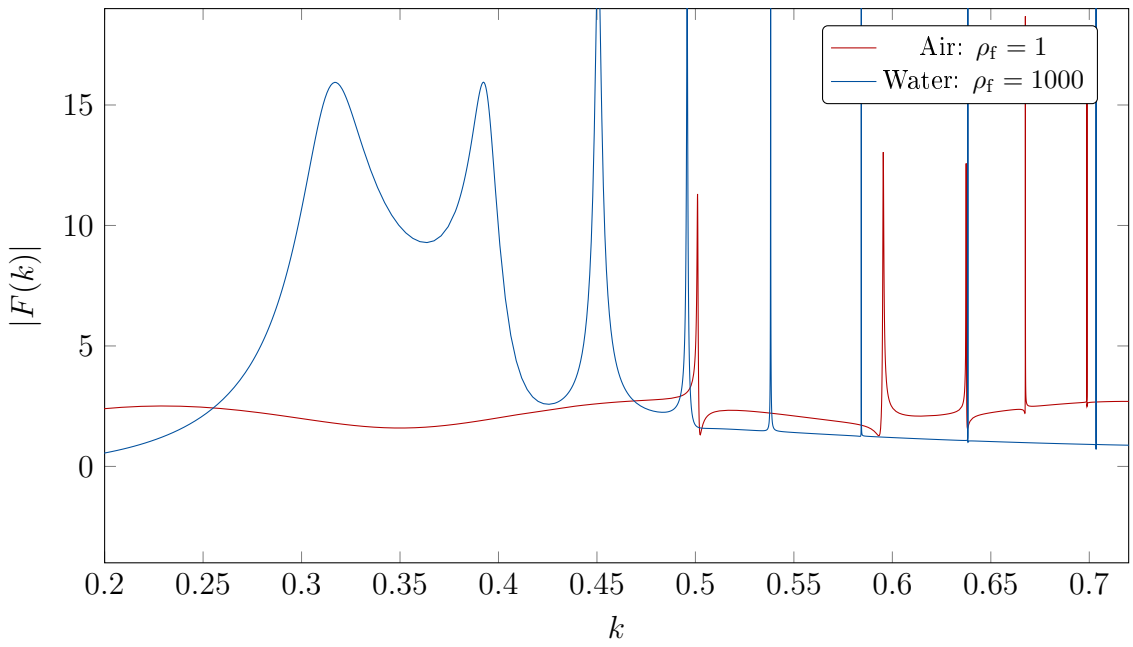


Figure 3.10: Scattering on elastic spherical shell: Scattering from elastic shell with different fluid densities.

uniform sampling, we would need several millions of samples. Due to the complexity of the functions involved, this would be very time consuming, so it is more practical to use non uniform sampling. Needless to say, to find these higher eigenmodes using a numerical approximation will be very hard, even though we know a priori the exact location of the analytic eigenmodes.

In Figure 3.10 we also show the effect of low density of the fluid. As we can see from Table 3.3, the eigenmodes of a spherical shell surrounded by air (with low density) comes close to the corresponding wave numbers for a spherical shell in vacuum. As Ihlenburg points out, an increase in the density of the acoustic medium around the spherical shell results in a larger shift to the left of the eigenmodes.

Ihlenburg uses a shell theory solution as a reference solution. The only difference will then be the function Z_n which in the case of shell theory is given by

$$Z_0 = \frac{Et\left[\Omega^2 - 2(1+nu)\right]}{\mathrm{i}\omega R^2\left(1-\nu^2\right)}$$

for $n = 0$ and

$$Z_n = \frac{\begin{vmatrix} \Omega^2 - (1+\beta^2)\left(\nu+\lambda_n-1\right) & -\beta^2(\nu+\lambda_n-1) - (1+\nu) \\ -\lambda_n\left(\beta^2(\nu+\lambda_n-1) + (1+\nu)\right) & \Omega^2 - 2(1+\nu) - \beta^2\lambda_n(\nu+\lambda_n-1) \end{vmatrix}}{-\mathrm{i}\omega \begin{vmatrix} \Omega^2 - (1+\beta^2)\left(\nu+\lambda_n-1\right) & 0 \\ -\lambda_n\left[\beta^2(\nu+\lambda_n-1)(1+\nu)\right] & -\frac{R^2\left(1-\nu^2\right)}{Et} \end{vmatrix}}$$

for $n > 0$, where

$$\Omega = \frac{R\omega}{c_s}, \quad \omega = kc_\mathrm{f}, \quad c_s = \sqrt{\frac{E}{(1+\nu^2)\rho_s}}, \quad \lambda_n = n(n+1) \quad \text{and} \quad \beta = \frac{t}{\sqrt{12}R}.$$

Table 3.3: Spherical shell: Comparison of wave number of modes in the case of air and water ($\rho_f = 1$ and $\rho_f = 1000$ respectively) and shall theory solution (for water) versus the values for spherical shell in vacuum.

Vacuum $\rho_f = 0$	Air $\rho_f = 1$	Water $\rho_f = 1000$	Shell theory $\rho_f = 1000$
0.501 391 438	0.501 076 117	0.317 068 140	0.319 125 000
0.595 778 716	0.595 394 549	0.392 380 087	0.394 382 813
0.637 906 314	0.637 483 918	0.450 626 523	0.453 015 747
0.667 867 024	0.667 543 974	0.495 802 270	0.498 512 170
0.699 070 510	0.698 803 667	0.538 194 517	0.541 388 924
0.738 130 214	0.737 894 055	0.584 225 861	0.588 206 393
0.789 033 751	0.788 813 951	0.638 340 506	0.643 567 469
0.854 204 203	0.853 992 114	0.703 448 660	0.710 555 398
0.934 884 353	0.934 674 041	0.781 278 648	0.791 080 189
1.031 426 209	1.031 213 325	0.872 642 564	0.886 139 492
1.143 580 741	1.152 864 073	0.977 700 431	0.996 080 715
1.270 753 897	1.270 526 694	1.096 200 472	1.120 841 376
1.412 195 450	1.411 393 912	1.227 664 075	1.260 134 201
1.567 117 288	1.566 867 299	1.371 508 743	1.413 569 088
1.734 756 714	1.734 492 945	1.527 120 122	1.580 722 768
1.914 404 195	1.914 125 251	1.693 889 193	1.761 172 291
2.105 411 065	1.944 795 610	1.871 228 418	1.954 506 268

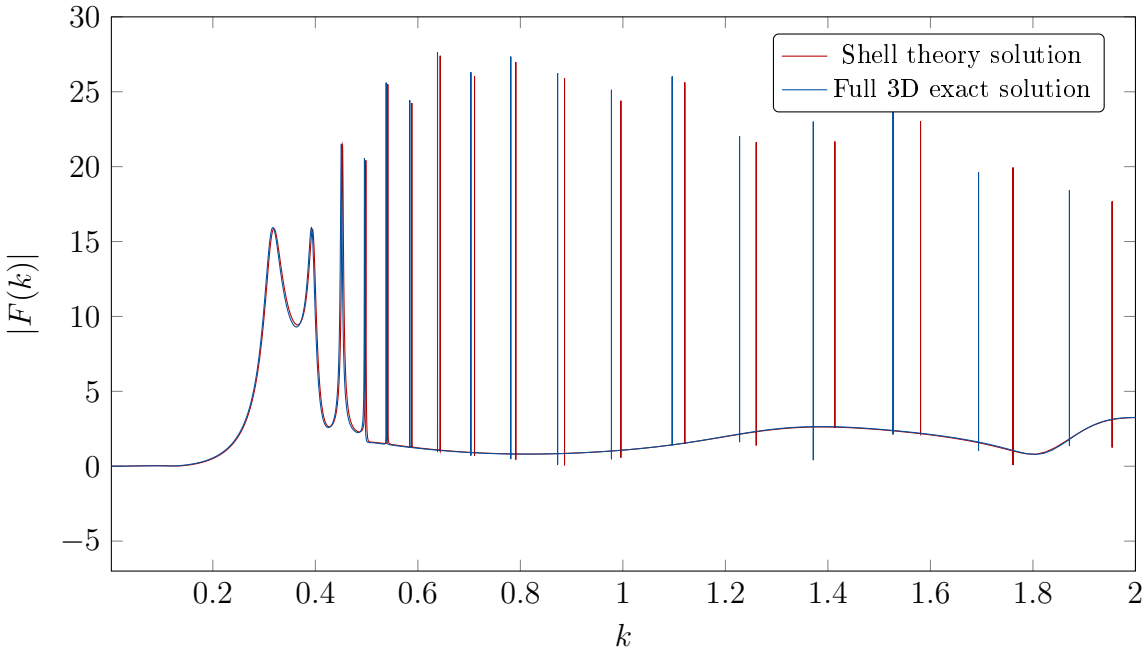


Figure 3.11: Scattering on elastic spherical shell: Comparison between shell theory solution and full 3D exact solution.

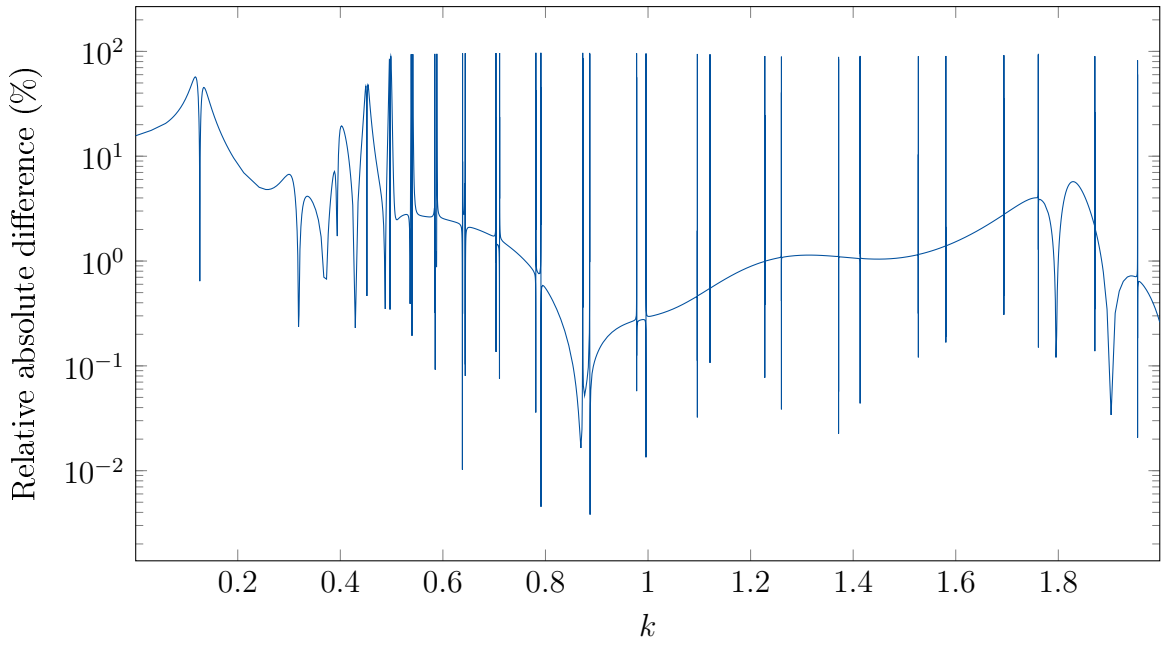


Figure 3.12: Scattering on elastic spherical shell: Error between shell theory solution and full 3D exact solution.

In Figure 3.11 we plot both our full 3D solution and this shell theory solution. The error is significant and is specially large at the eigenmodes. In Figure 3.12 a more informative picture regarding the error is computed. Due to the large variation of the data sets, the most reasonable norm is the absolute difference divided by the maximum absolute value. That is, for two finite data sets $\{x_i\}$ and $\{y_i\}$ the relative difference E_i for each pair (x_i, y_i) is given by

$$E_i = \frac{|x_i - y_i|}{\max(|x_i|, |y_i|)} \tag{3.3}$$

We obviously also here see large errors at every eigenmode, but also elsewhere, the error is significant. In average, the error lies above 1%. In the applications of complex geometries, this result would be acceptable, but we want to be able to perform convergence analysis on this spherical shell where the obtained error may end up being far below 1%. This way, we can draw conclusion of which sets of parameters increases the convergence rate the most. The exactness of the solution is therefore considered to be of importance.

Chapter 4

Isogeometric analysis

Isogeometric analysis (IGA), is basically an extension of the finite element method (FEM) using non-uniform rational B-splines (NURBS) as basis functions. Being introduced in 2005 by Hughes et al. [1], followed by the book [2] in 2009, IGA tries to bridge the gap between finite element analysis and CAD design tools. The important future of IGA is that it uses the same basis as CAD software for describing the given geometry, and thus exact representation of the model is possible. It is therefore natural to include a section considering this basis in the beginning before we set up the IGA for the problem at hand. Note that only the relevant parts of the IGA will be presented, and even here we shall be brief. This chapter assumes the reader is somewhat familiar which the finite element method.

4.1 B-splines

The NURBS basis is constructed by B-splines. So an understanding of B-splines is crucial to understanding NURBS. Let p be the polynomial order, let n be the number of basis functions and let $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$ be an ordered vector with non decreasing elements, called *knots* (referred to as a *knot vector*). Then, the n B-splines, $\{N_{i,p}\}_{i\in[1,n]}$, are recursively defined by

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

where

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{if } \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

This formula is referred to as Cox-deBoor formula, and the derivative of a spline may be computed by

$$\frac{\mathrm{d}}{\mathrm{d}\xi} N_{i,p}(\xi) = \frac{p}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) - \frac{p}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi). \tag{4.2}$$

Throughout the project we shall use *open knot vectors*. That is, the first and last element in the vector is repeated $p + 1$ times. Moreover, we shall use *normalized* knot vectors,

which simply spans from 0 to 1. Finally, a knot is said to have *multiplicity* m if it is repeated m times in Ξ.

Some important properties of B-splines are given by the following (for proof, cf. [15]).

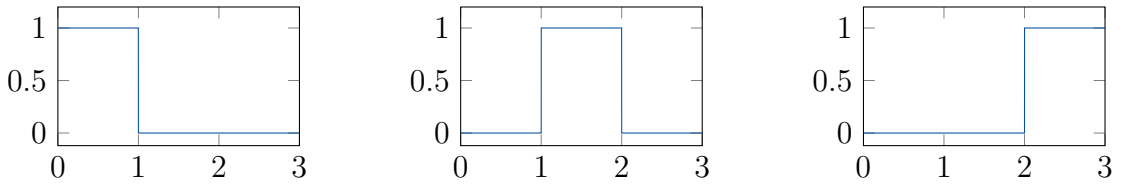
1. $N_{i,p}$ are piecewise polynomials.
2. $N_{i,p}$ depends only on the knots $\xi_i, \xi_{i+1}, \ldots, \xi_{i+p+1}$.
3. In general $N_{i,p}(\xi) \geq 0$, and if $\xi \notin [\xi_i, \xi_{i+p+1})$ then $N_{i,p}(\xi) = 0$.
4. If $\xi \in [\xi_\mu, \xi_{\mu+1})$ then $N_{i,p}(\xi) = 0$ if $i < \mu - p$ or $i > \mu$.
5. If $\xi \in (\xi_i, \xi_{i+p+1})$ then $N_{i,p} > 0$.
6. If $\bar{\xi} = \xi_{j+1} = \cdots = \xi_{j+p} < \xi_{j+p+1}$ then $N_{i,p}(\bar{\xi}) = \delta_{ij}$.
7. If a knot $\bar{\xi} \in \{\xi_i, \ldots, \xi_{i+p+1}\}$ has multiplicity m then $N_{i,p}$ is $p - m$ differentiable at $\bar{\xi}$.
8. B-splines satisfies the partition of unity property. That is,
$$\sum_{i=1}^{n} N_{i,p}(\xi) = 1 \qquad \forall \xi, p.$$
9. B-splines forms a *stable* basis for piecewise polynomials.

To ease the understanding of B-splines we shall construct an illustrative example. Consider quadratic B-splines ($p = 2$) with the knot vector $\Xi = \{0, 0, 0, 1, 2, 2, 3, 3, 3\}$. Note that the use of a non-normalized knot vector is only for convenience and carries no importance; the B-splines would have had the same characteristics if we divided all knots by 3. Since $|\Xi| = 10$, the number of basis functions is given by $n = |\Xi| - p - 1 = 6$. In Figure 4.1, Figure 4.2 and Figure 4.3, we have plotted not only the 6 basis functions of second order, but also the functions of order zero and one needed to evaluate these basis functions. By property 7 in the previous list, we see that $N_{1,2}$ and $N_{6,2}$ are discontinuous at $\xi = 0$ and $\xi = 3$, respectively ($p = 2$ and $m = 3$ yields C^{-1} continuity in the endpoints). This is characteristic for all open knot vectors. Also note that the repeated knot at $\xi = 2$ forces the function to have the Kronecker delta property; that is, $N_{i,p}(\xi_j) = \delta_{ij}$ if ξ_j has multiplicity $m = p$.

We may now define a *spline curve* by

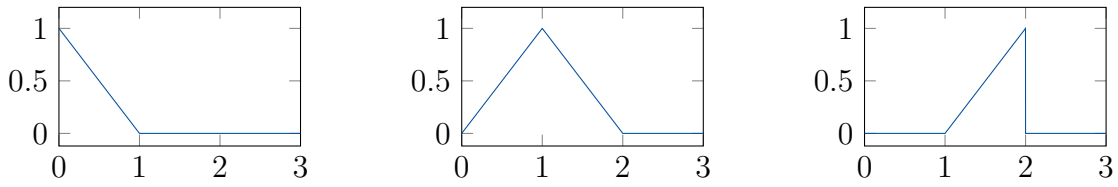
$$\boldsymbol{P}(\xi) = \sum_{i=1}^{n} N_{i,p}(\xi) \boldsymbol{P}_i$$

where $\{\boldsymbol{P}_i\}_{i \in [1,n]}$ are the *control points* of the curve. To continue the example, consider the control points $\boldsymbol{P}_1 = (5, 3)$, $\boldsymbol{P}_2 = (3, 2)$, $\boldsymbol{P}_3 = (1, 7)$, $\boldsymbol{P}_4 = (4, 3)$, $\boldsymbol{P}_5 = (7, 6)$ and $\boldsymbol{P}_6 = (7, 1)$. Using the same basis functions $\{N_{i,2}\}_{i \in [1,6]}$ as in the previous example, we get the curve depicted in Figure 4.4. In addition to the curve, the *control polygon* is drawn, which is simply the piece wise linear curve between the ordered control points. Note that the smoothness of the curve degrades at the point $(4, 3)$. This is the result of the repeated knot $\xi = 2$ which yields an interpolation effect, such that the control point lies on the curve (which is also the case at the end points). Moreover, the control polygon is tangential at all knots.

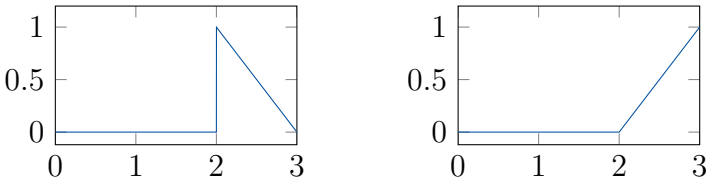


(a) Plot of $N_{3,0}$. (b) Plot of $N_{4,0}$. (c) Plot of $N_{6,0}$.

Figure 4.1: Plot of the non-zero B-splines of order zero, where the knot vector is given by $\Xi = \{0, 0, 0, 1, 2, 2, 3, 3, 3\}$. Note that $N_{j,0} \equiv 0$ for $j \in \{1, 2, 5, 6, 7, 8\}$.

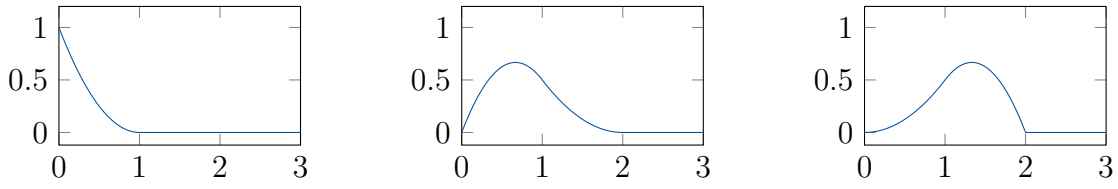


(a) Plot of $N_{2,1}$. (b) Plot of $N_{3,1}$. (c) Plot of $N_{4,1}$.

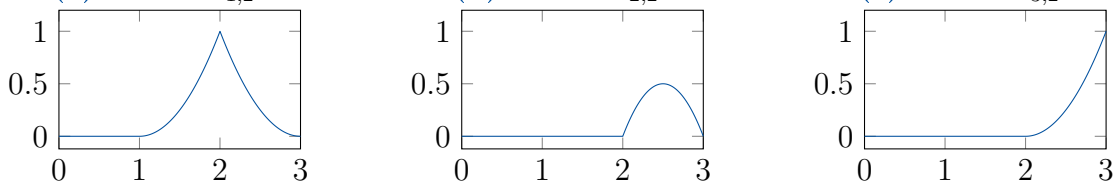


(d) Plot of $N_{5,1}$. (e) Plot of $N_{6,1}$.

Figure 4.2: Plot of the non-zero B-spline basis functions (of first degree), where the knot vector is given by $\Xi = \{0, 0, 0, 1, 2, 2, 3, 3, 3\}$. Note that $N_{j,1} \equiv 0$ for $j \in \{1, 7\}$.

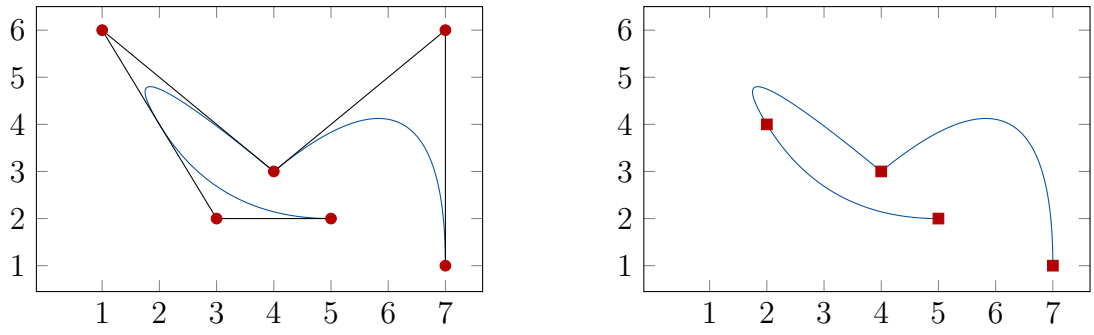


(a) Plot of $N_{1,2}$. (b) Plot of $N_{2,2}$. (c) Plot of $N_{3,2}$.



(d) Plot of $N_{4,2}$. (e) Plot of $N_{5,2}$. (f) Plot of $N_{6,2}$.

Figure 4.3: Plot of the 6 different B-spline basis functions (of second degree), where the knot vector is given by $\Xi = \{0, 0, 0, 1, 2, 2, 3, 3, 3\}$.



(a) Curve and control polygon. **(b)** Curve and location of knots.

Figure 4.4: Plot of a spline curve. Since $\boldsymbol{P}_1 = (5, 2)$, this is the point where the curve starts ($\xi = 0$).

4.1.1 B-spline knot insertion

Knot insertion is a process for which knots are inserted into the knot vector, which would create more basis function without changing the geometry. This is a very important concept as it allows us to enrich the basis without changing the geometry. Thus, knot insertion corresponds to the classical h-refinement procedure in FEM (refining the mesh). This is because the knot vector defines the mesh; such that when more knots are inserted, we get a more refined mesh.

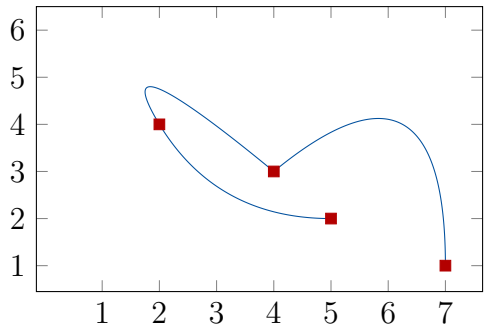
The goal is to insert more knots into Ξ without changing the shape of the curve. We shall do this by *Böhm's method* (cf. [15]) which does knot insertion by inserting one knot at a time. Let Ξ' be the new knot vector after a knot have been inserted into Ξ. Let $\{N_{i,p,\Xi}\}_{i\in[1,n]}$ be the old basis (corresponding to Ξ) and $\{N_{i,p,\Xi'}\}_{i\in[1,n+1]}$ the new basis after a knot has been inserted (note that since Ξ' is known, the basis is completely determined). We then want to find the new set of control points $\{\boldsymbol{P}'_i\}_{i\in[1,n+1]}$ (here $\boldsymbol{P}'_i \in \mathbb{R}^d$ where d is the dimension of the space for which the spline curve belongs) such that

$$\boldsymbol{P}(\xi) = \sum_{i=1}^{n} N_{i,p,\Xi}(\xi)\boldsymbol{P}_i \overset{!}{=} \sum_{i=1}^{n+1} N_{i,p,\Xi'}(\xi)\boldsymbol{P}'_i.$$

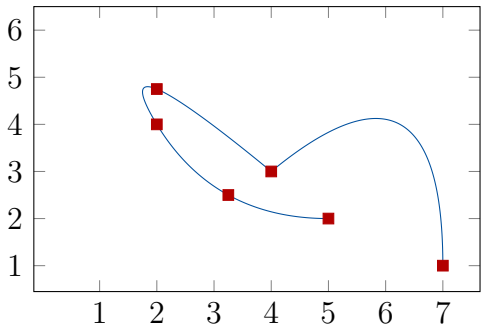
This results in a linear system of equation which could be solved by brute force. However, Böhm method exploits the support property of B-splines to improve efficiency. Assume that the new knot $\bar{\xi}$ is inserted in the interval $[\xi_\mu, \xi_{\mu+1}]$. Then

$$\boldsymbol{P}'_i = \begin{cases} \boldsymbol{P}_i, & \text{if } 1 \le i \le \mu - p \\ \frac{\bar{\xi}-\xi_i}{\xi_{i+p}-\xi_i}\boldsymbol{P}_i + \frac{\xi_{i+p}-\bar{\xi}}{\xi_{i+p}-\xi_i}\boldsymbol{P}_{i-1}, & \text{if } \mu - p + 1 \le i \le \mu \\ \boldsymbol{P}_{i-1}, & \text{if } \mu + 1 \le i \le n + 1. \end{cases}$$

Let's continue our example by inserting the knots $\xi = 0.5$ and $\xi = 1.5$ into our knot vector. We then simply use Böhms method twice to calculate the new control points. The curve with the mesh before and after the knot insertion is depicted in Figure 4.5. In Figure 4.6 we see that the control polygon has changed. Indeed, it has moved towards the curve where the refinement has occurred, which is an other nice property of knot insertion.

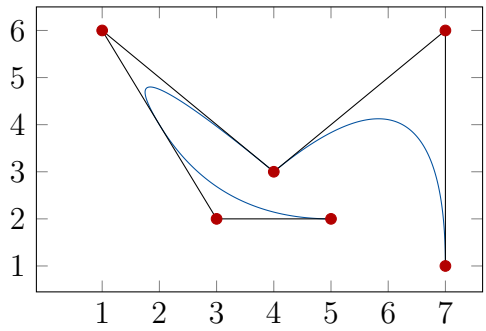


(a) Curve with mesh before refinement.

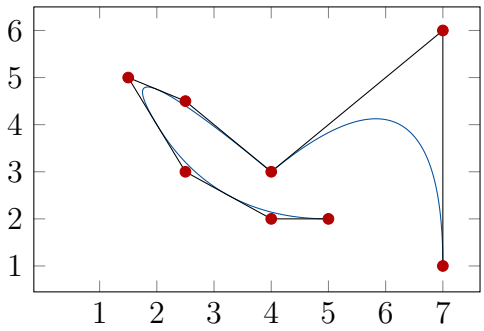


(b) Curve with mesh after refinement.

Figure 4.5: Mesh comparison for knot refinement.



(a) Curve and control polygon before refinement.



(b) Curve and control polygon after refinement.

Figure 4.6: Control polygon comparison for knot insertion.

4.1.2 B-spline degree elevation

Having a basis of a higher order creates more accurate solution in FEM/IGA. Thus, we want to have an algorithm which increase the order from p to $p+m$ without changing the geometry and the parametric space. Since the continuity at each knot must be preserved, it follows from property 7 on page 28 that we must increase the multiplicity of each knot by m. We also need to find the new set of control points. As for knot insertion, we must find the new set of control points $\{\boldsymbol{Q}_i\}_{i\in[1,n']}$ such that

$$\boldsymbol{P}(\xi) = \sum_{i=1}^{n} N_{i,p,\Xi}(\xi)\boldsymbol{P}_i \overset{!}{=} \sum_{i=1}^{n'} N_{i,p+m,\Xi'}(\xi)\boldsymbol{Q}_i =: \boldsymbol{Q}(\xi).$$

Let S be an integer such that $S+1$ is the number of unique knots in Ξ. Since Ξ is open ($\xi_1 = \cdots = \xi_{p+1}$ and $\xi_{n+1} = \cdots = \xi_{n+p+1}$), it may be written on the form

$$\begin{aligned}\Xi &= \{\underbrace{\xi_1,\ldots,\xi_{p+1}}_{p+1},\xi_{p+2},\ldots,\xi_n,\underbrace{\xi_{n+1},\ldots,\xi_{n+p+1}}_{p+1}\} \\ &= \{\underbrace{u_0,\ldots,u_0}_{p+1},\underbrace{u_1,\ldots,u_1}_{z_1},\ldots,\underbrace{u_{S-1},\ldots,u_{S-1}}_{z_{S-1}},\underbrace{u_S,\ldots,u_S}_{p+1}\}\end{aligned}$$

where z_i denotes the multiplicity of the knot with value u_i for $i = 1, \dots, S-1$. If we now want to elevate the degree m times (from p to $p+m$), we get the new knot vector

$$\begin{aligned}\Xi' &= \{\underbrace{\xi'_1, \dots, \xi'_{p+1}}_{p+1+m}, \xi'_{p+2}, \dots, \xi'_{n'}, \underbrace{\xi'_{n'+1}, \dots, \xi'_{n'+p+1+m}}_{p+1+m}\} \\ &= \{\underbrace{u_0, \dots, u_0}_{p+1+m}, \underbrace{u_1, \dots, u_1}_{z_1+m}, \dots, \underbrace{u_{S-1}, \dots, u_{S-1}}_{z_{S-1}+m}, \underbrace{u_S, \dots, u_S}_{p+1+m}\}.\end{aligned}$$

It is also easy to observe that the new number of basis functions is given by $n' = n + S \cdot m$. What remains to be found is the new set of control points $\{\boldsymbol{Q}_i\}_{i\in[1,n']}$. Several efficient algorithms exist for this purpose, but we shall follow the idea presented by Huang2004ede et al. in [16]. Since the notation of the article does not correspond to the notation presented here, a complete derivation of the algorithm will be presented here. Denote by $\boldsymbol{P}^{(l)}$ the l'th derivative of the spline curve of degree p (which will have degree $p-l$) such that (using an inductive argument and Equation (4.2))

$$\boldsymbol{P}^{(j)}(\xi) = \sum_{i=1}^{n-j} N_{i+l,p-j,\Xi}(\xi)\boldsymbol{P}_i^j \quad \text{and} \quad \boldsymbol{Q}^{(j)}(\xi) = \sum_{i=1}^{n'-j} N_{i+j,p+m-j,\Xi'}(\xi)\boldsymbol{Q}_i^j$$

where the coefficients $\boldsymbol{P}_i^j$ are defined recursively by

$$\boldsymbol{P}_i^j = \begin{cases} \frac{p+1-j}{\xi_{i+p+1}-\xi_{i+j}}\left(\boldsymbol{P}_{i+1}^{j-1} - \boldsymbol{P}_i^{j-1}\right) & \text{if } \xi_{i+p+1} > \xi_{i+j} \\ \boldsymbol{0} & \text{if } \xi_{i+p+1} = \xi_{i+j} \end{cases} \tag{4.3}$$

for $j > 0$, starting with $\boldsymbol{P}_i^0 = \boldsymbol{P}_i$ for $j = 0$. Correspondingly we have,

$$\boldsymbol{Q}_i^j = \begin{cases} \frac{p+m+1-j}{\xi'_{i+p+m+1}-\xi'_{i+j}}\left(\boldsymbol{Q}_{i+1}^{j-1} - \boldsymbol{Q}_i^{j-1}\right) & \text{if } \xi'_{i+p+m+1} > \xi'_{i+j} \\ \boldsymbol{0} & \text{if } \xi'_{i+p+m+1} = \xi'_{i+j}. \end{cases} \tag{4.4}$$

Note that this implies the following useful formula we shall use later

$$\boldsymbol{Q}_{i+1}^{j-1} = \boldsymbol{Q}_i^{j-1} + \frac{\xi'_{i+1+p+m} - \xi'_{i+j}}{p+m+1-j}\boldsymbol{Q}_i^j \qquad \text{if } \xi'_{i+p+m+1} > \xi'_{i+j}. \tag{4.5}$$

Since $u_0 = \xi_{j+1} = \xi_{j+1+1} = \cdots = \xi_{j+1+p-j} < \xi_{p+2}$ for $0 \le j \le p$, property 6 on page 28 implies that

$$N_{i+j,p-j,\Xi}(u_0) = \delta_{i+j,j+1}.$$

Hence,

$$\boldsymbol{P}^{(j)}(u_0) = \sum_{i=1}^{n-j} N_{i+j,p-j,\Xi}(u_0)\boldsymbol{P}_i^j = \sum_{i=1}^{n-j} \delta_{i+j,l+1}\boldsymbol{P}_i^j = \boldsymbol{P}_1^j.$$

We have a corresponding result for $\boldsymbol{Q}^{(j)}$ such that

$$\boldsymbol{P}^{(j)}(u_0) = \boldsymbol{P}_1^j \quad \text{and} \quad \boldsymbol{Q}^{(j)}(u_0) = \boldsymbol{Q}_1^j.$$

Moreover, $\boldsymbol{P}(\xi)$ and $\boldsymbol{Q}(\xi)$ have the same geometry and parameterization ($\boldsymbol{P}^{(j)}(\xi) = \boldsymbol{Q}^{(j)}(\xi)$), and we must therefore have

$$\boldsymbol{Q}_1^j = \boldsymbol{P}_1^j \tag{4.6}$$

for $0 \leq j \leq p$.

Define

$$\beta_i = \sum_{l=1}^{i} z_l,$$

such that we have $u_i = \xi_{\beta_i+p+1}$ (the last of the repeated knot). Let $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$, such that we only consider the case when the degree of $\boldsymbol{P}^{(j)}$ satisfy $p-j \leq z_i - 1$. Since the knot u_i is repeated z_i times, property 6 on page 28 now implies that

$$N_{\hat{i},p-j,\Xi}(u_i) = \delta_{\hat{i},\beta_i+1+j},$$

such that

$$\begin{aligned}\boldsymbol{P}^{(j)}(u_i) &= \sum_{\hat{i}=1}^{n-j} N_{\hat{i}+j,p-j,\Xi}(u_i)\boldsymbol{P}^j_{\hat{i}} = \sum_{\hat{i}=1}^{n-j} \delta_{\hat{i}+j,\beta_i+1+j}\boldsymbol{P}^j_{\hat{i}} \\ &= \boldsymbol{P}^j_{\beta_i+1}\end{aligned}$$

for $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$.

We note that $u_i = \xi'_{\beta_i+p+1+m+im}$ and that this knot has multiplicity $z_i + m$ in Ξ'. Again we consider the indices i and j to satisfy $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$, such that we only consider the case when the degree of $\boldsymbol{Q}^{(j)}$ satisfy $p+m-j \leq z_i + m - 1$. Using once again property 6 on page 28 we have

$$N_{\hat{i},p+m-j,\Xi'}(u_i) = \delta_{\hat{i},\beta_i+1+j},$$

such that

$$\begin{aligned}\boldsymbol{Q}^{(j)}(u_i) &= \sum_{\hat{i}=1}^{n'-j} N_{\hat{i}+j,p+m-j,\Xi'}(u_i)\boldsymbol{Q}^j_{\hat{i}} = \sum_{\hat{i}=1}^{n'-j} \delta_{\hat{i}+j,\beta_i+1+j}\boldsymbol{Q}^j_{\hat{i}} \\ &= \boldsymbol{Q}^j_{\beta_i+1}\end{aligned}$$

for $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$. Using the fact that $\boldsymbol{P}^{(j)}(\xi) = \boldsymbol{Q}^{(j)}(\xi)$, we have obtained the formula

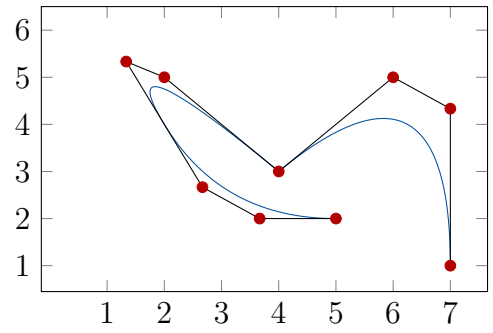
$$\boldsymbol{Q}^j_{\beta_i+1+im} = \boldsymbol{P}^j_{\beta_i+1}, \qquad p+1-z_i \leq j \leq p, \quad 1 \leq i \leq S-1. \tag{4.7}$$

Since $\boldsymbol{P}(\xi)$ has degree p it's $(p+1)$'th derivative must be zero, and thus also the $(p+1)$'th derivative of $\boldsymbol{Q}(\xi)$. Using property 4 on page 28 and letting $\xi \in [u_i, u_{i+1}) = [\xi'_{\beta_i+p+1+m+im}, \xi'_{\beta_i+p+1+m+im+1})$, we have

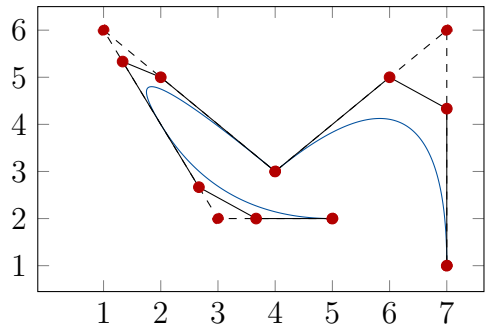
$$\begin{aligned}\mathbf{0} = \boldsymbol{P}^{(p+1)}(\xi) = \boldsymbol{Q}^{(p+1)}(\xi) &= \sum_{\hat{i}=1}^{n'-(p+1)} N_{\hat{i}+p+1,m-1,\Xi'}(\xi)\boldsymbol{Q}^{p+1}_{\hat{i}} \\ &= \sum_{\hat{i}=\beta_i+1+im}^{\beta_i+m+im} N_{\hat{i}+p+1,m-1,\Xi'}(\xi)\boldsymbol{Q}^{p+1}_{\hat{i}},\end{aligned}$$

which implies, that

$$\boldsymbol{Q}^{p+1}_{\hat{i}} = \mathbf{0}, \qquad \beta_i + 1 + im \leq \hat{i} \leq \beta_i + m + im,$$



(a) Curve and control polygon after degree elevation.



(b) Control polygon comparison showing the corner cutting effect.

Figure 4.7: Control polygon comparison for degree elevation of a spline curve.

for $0 \leq i \leq S-1$. Using Equation (4.5) with $j = p+1$, we therefore have

$$\boldsymbol{Q}^p_{\hat{i}+1} = \boldsymbol{Q}^p_{\hat{i}}, \qquad \beta_i + 1 + im \leq \hat{i} \leq \beta_i + m + im, \quad 0 \leq i \leq S-1,$$

or equivalently

$$\boldsymbol{Q}^p_{\beta_i+1+im+k} = \boldsymbol{Q}^p_{\beta_i+1+im}, \qquad 1 \leq k \leq m, \quad 0 \leq i \leq S-1. \tag{4.8}$$

In addition, we see from Equation (4.4) that $\boldsymbol{Q}^p_{\hat{i}} = \boldsymbol{0}$ whenever $\xi'_{\hat{i}+p+m+1} = \xi'_{\hat{i}+p}$. Since the knots u_{i+1} are repeated $z_{i+1}+m$ times in Ξ' (starting at $\xi'_{\beta_i+1+p+im+m+1}$) we have

$$\boldsymbol{Q}^p_{\beta_i+1+im+k} = \boldsymbol{0}, \qquad m+1 \leq k \leq m-1+z_{i+1}, \quad 0 \leq i \leq S-1.$$

It turns out that these points are not needed.

So in summary, the algorithm does the following steps

1. Set $\boldsymbol{P}^0_i = \boldsymbol{P}_i$ for $1 \leq i \leq n$.
2. Compute $\boldsymbol{P}^j_1$ for $0 \leq j \leq p$ and $\boldsymbol{P}^j_{\beta_i+1}$ for $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$ using Equation (4.3).
3. Compute $\boldsymbol{Q}^j_1$ for $0 \leq j \leq p$ using Equation (4.6)
4. Compute $\boldsymbol{Q}^j_{\beta_i+1+im}$ for $p+1-z_i \leq j \leq p$ and $1 \leq i \leq S-1$ using Equation (4.7).
5. Compute $\boldsymbol{Q}^p_{\beta_i+1+im+k}$ for $1 \leq k \leq m$ and $0 \leq i \leq S-1$ using Equation (4.8).
6. Compute the new control points $\boldsymbol{Q}_i = \boldsymbol{Q}^0_i$ backwards from Equation (4.5).

This algorithm may be optimized as discussed in [16], but will not be presented here as the efficiency of this algorithm is not of great importance for this thesis.

Continuing on our example, let's elevate the degree of the original spline curve by one (from 2 to 3). As we can see in Figure 4.7a, the geometry of the curve has not changed but the control polygon has changed. As we can see in Figure 4.7b only non-interpolating control points in the control polygon have changed. Two new control points replaces each of these control points such that we get a corner cutting effect in the control polygon. This is why degree elevation of spline curves is called corner cutting.

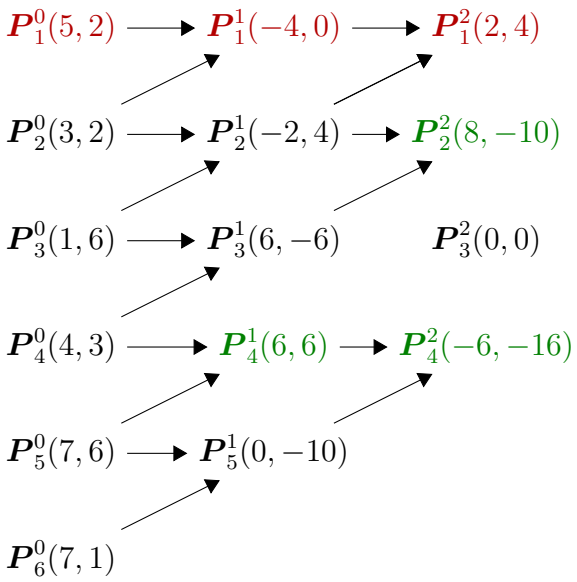


Figure 4.8: Graph of differential coefficients of $\boldsymbol{P}(\xi)$ in the example.

In Figure 4.8 we have a graph which illustrates this process. The points which are colored are used directly to compute coefficients in the set $\{\boldsymbol{Q}_i^j\}_{i,j}$ (which have corresponding colors in the graph in Figure 4.9). Except for $\boldsymbol{P}_3^2(0,0)$ (which is set to zero because $\xi_6 = \xi_5$) all points are thus needed.

In Figure 4.9 we have the next graph where we illustrate how the differential coefficients of $\boldsymbol{Q}(\xi)$ in red are first computed. The coefficients in red computed (step 3), the coefficients in green are then computed (step 4) such that the coefficients in yellow may be computed (step 5). The rest of the coefficients are computed by Equation (4.5). Note that $\boldsymbol{Q}_5^2$ is not really needed.

4.1.3 Spline volumes

The extension to bivariate spline surfaces and trivariate spline volumes is straight forward (we shall only consider volumes). Let $\{N_{i,p}\}_{i\in[1,n]}$, $\{M_{j,q}\}_{j\in[1,m]}$ and $\{L_{k,p}\}_{k\in[1,l]}$ be the set of B-spline basis functions in ξ-, η- and ζ-direction, respectively. These set's have their own degree (p, q and r, respectively) and knot vectors (Ξ, $\mathcal{H}$ and $\mathcal{Z}$, respectively). A spline volume is then defined by

$$\boldsymbol{V}(\xi,\eta,\zeta) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p}(\xi)M_{j,q}(\eta)L_{k,r}(\zeta)\boldsymbol{P}_{i,j,k}.$$

We may extend our spline curve example into a volume by adding the knot vectors $\mathcal{H} = \{0,0,1,1\}$ and $\mathcal{Z} = \{0,0,1,1\}$. By adding appropriate control points we get the spline volume in Figure 4.10. We have here in addition to the volume drawn the mesh on top showing three elements.

Once again, we want to be able to refine this mesh into more elements without changing the geometry. For spline volumes, this is done by refining the mesh in each parameter

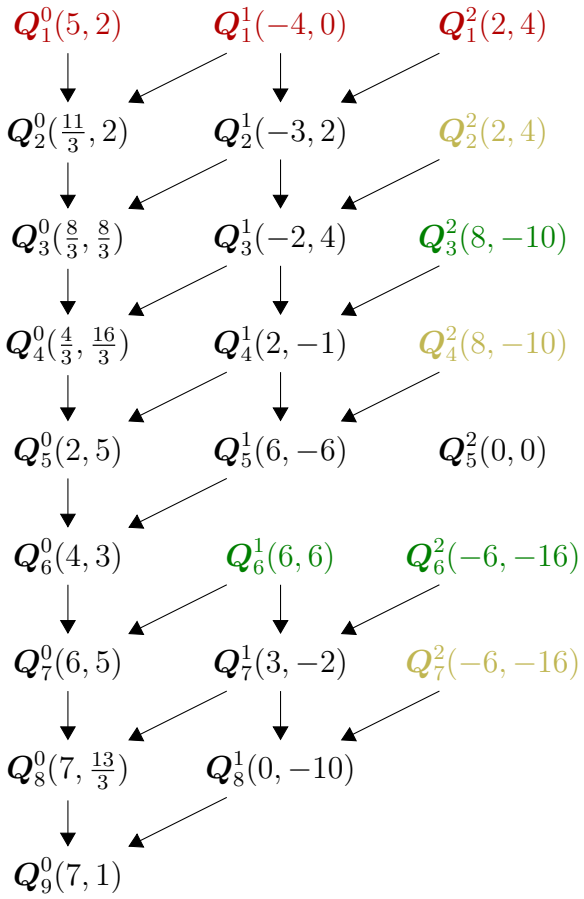
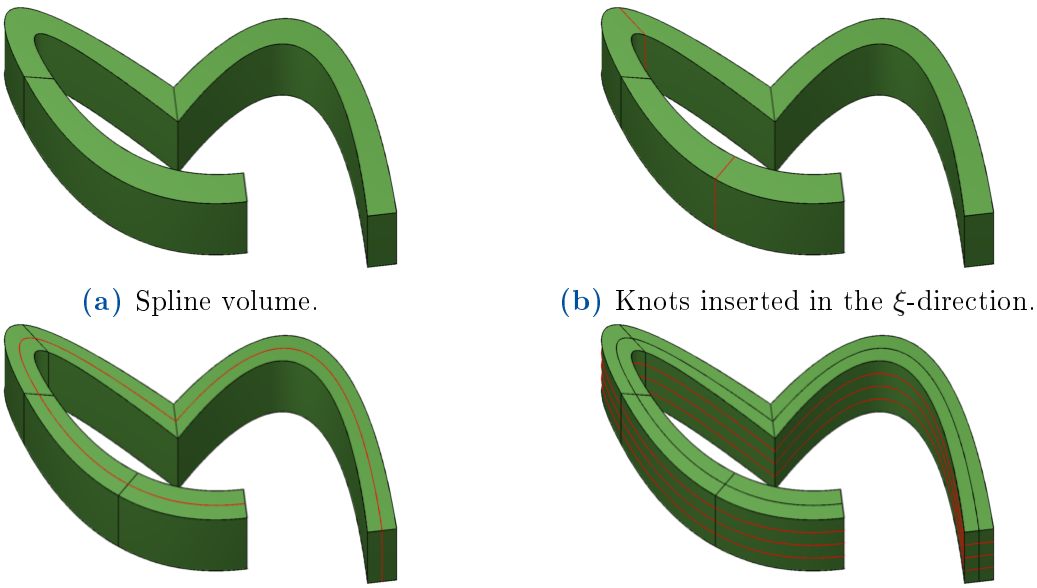


Figure 4.9: Differential coefficients of $\boldsymbol{Q}(\xi)$ in the example.



(a) Spline volume.

(b) Knots inserted in the ξ-direction.

(c) Knots inserted in the η-direction.

(d) Knots inserted in the ζ-direction.

Figure 4.10: Spline volume and its knot insertions. We here choose to first insert knots in the ξ-direction followed by the η- and ζ-direction, respectively.

direction (in the ξ-, η- and ζ-direction). This refining process thus involves refining in each direction separately. For each knot vector, we do knot insertion by Böhms method once again. Here, we have to order the structure of the control points $\{\boldsymbol{P}_{i,j,k}\}$ in a specific way. As an example, say we want to insert s knots in the Ξ direction for a spline object living in a d-dimensional space using Böhms method. We then want to find the new set of control points $\{\boldsymbol{P}'_{i,j,k}\}$ such that

$$\begin{aligned}\boldsymbol{V}(\xi,\eta,\zeta) &= \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p,\Xi}(\xi)M_{j,q,\mathcal{H}}(\eta)L_{k,r,\mathcal{Z}}(\zeta)\boldsymbol{P}_{i,j,k} \\ &\overset{!}{=} \sum_{i=1}^{n+s}\sum_{j=1}^{m}\sum_{k=1}^{l} N_{i,p,\Xi'}(\xi)M_{j,q,\mathcal{H}}(\eta)L_{k,r,\mathcal{Z}}(\zeta)\boldsymbol{P}'_{i,j,k}.\end{aligned}$$

We shall use the ordering $\bar{\boldsymbol{P}}_i = \{\boldsymbol{P}_{i,1,1}, \boldsymbol{P}_{i,2,1}, \ldots, \boldsymbol{P}_{i,m,1}, \boldsymbol{P}_{i,1,2}, \ldots, \boldsymbol{P}_{i,m,l}\}$. We create a new fictitious spline curve given by

$$\bar{\boldsymbol{C}}(\xi) = \sum_{i=1}^{n} N_{i,p,\Xi}(\xi)\bar{\boldsymbol{P}}_i \overset{!}{=} \sum_{i=1}^{n+s} N_{i,p,\Xi'}(\xi)\bar{\boldsymbol{P}}'_i$$

where $\bar{\boldsymbol{P}}'_i$ is to be determined using Böhms method s times. Note that this spline curve is in a $(d\cdot m\cdot l)$-dimensional space. Converting $\{\bar{\boldsymbol{P}}'_i\}$ back to the old structure, we obtain the resulting $\{\boldsymbol{P}'_{i,j,k}\}$. Similar procedure is done if knots are inserted in η- and ζ-direction.

Let's say we want to insert the knots $\{0.5, 1.5\}$ in the ξ-direction (as we did for the spline curve) and the knots $\{0.5\}$ and $\{0.25, 0.5, 0.75\}$ for the η- and ζ-direction, respectively. The result (in Figure 4.10) is a refined mesh from 3 elements to 40 elements.

A corresponding procedure is done for degree elevation in spline volumes.

4.2 NURBS

With the B-splines in our arsenal, we are ready to present Non-Uniform Rational B-Splines (NURBS). All though B-splines may represent many complex curves, there are a class of curves which may not be represented exactly by B-splines, namely, conic sections like circles. Such shapes are often used in engineering, and thus, an extension which enables this would be valuable. NURBS enables us to tackle such geometries as well.

Let $\{w_i\}_{i\in[1,n]}$ be a set of *weights*, and define the *weighting function* by

$$W(\xi) = \sum_{\hat{i}=1}^{n} N_{\hat{i},p}(\xi)w_{\hat{i}}.$$

The one dimensional NURBS basis functions can now be defined by

$$R_{\mathrm{i}}^{p}(\xi) = \frac{N_{i,p}(\xi)w_i}{W(\xi)}$$

such that a NURBS curve may be expressed by

$$\boldsymbol{C}(\xi) = \sum_{i=1}^{n} R_{\mathrm{i}}^{p}(\xi)\boldsymbol{P}_i.$$

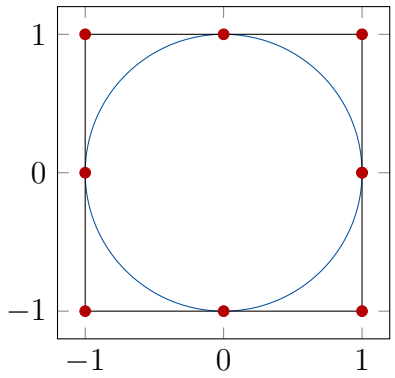


Figure 4.11: NURBS representation of a unit circle with the corresponding control polygon. The curve goes counter clockwise around the unit circle (which is a result of the corresponding ordering in the control points), starts at $(1, 0)$ and ends at $(1, 0)$.

There are several ways to construct a circle using NURBS (but we need $p \geq 2$). For example, consider the knot vector $\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$, the weights given by $\{w_i\}_{i\in[1,9]}$ where

$$w_i = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } i \text{ even} \\ 1 & \text{otherwise.} \end{cases}$$

and the control points $\boldsymbol{P}_1 = (0, 0)$, $\boldsymbol{P}_2 = (1, 1)$, $\boldsymbol{P}_3 = (0, 1)$, $\boldsymbol{P}_4 = (-1, 1)$, $\boldsymbol{P}_5 = (-1, 0)$, $\boldsymbol{P}_6 = (-1, -1)$, $\boldsymbol{P}_7 = (0, -1)$, $\boldsymbol{P}_8 = (1, -1)$ and $\boldsymbol{P}_9 = (1, 0)$. This will produce the unit circle which is depicted in Figure 4.11. Note that there is no non-repeated knot in the knot vector, so we will have interpolation between the location of the knots and the control polygon.

4.2.1 NURBS knot insertion

A d-dimensional NURBS curve is a projection of a $(d+1)$-dimensional B-spline curve (cf. [2]). We may exploit this property to insert new knots into a NURBS.

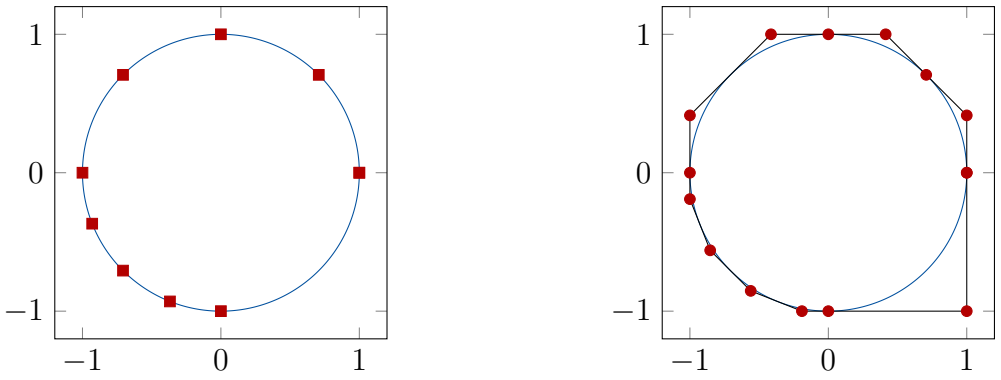
Let's say we want to insert s new knots into a NURBS curve defined by the knot vector Ξ, the control points $\{\boldsymbol{P}_i\}_{i\in[1,n]} = \{(x_i, y_i)\}_{i\in[1,n]}$, and the weights $\{w_i\}_{i\in[1,n]}$. We then construct a 3D B-spline curve with control points given by

$$\{\boldsymbol{Q}_i\}_{i\in[1,n]} = \{(w_i x_i, w_i y_i, w_i)\}_{i\in[1,n]}\,.$$

We now have a B-spline curve in 3D defined by Ξ (the same knot vector) and the control points $\{\boldsymbol{Q}_i\}_{i\in[1,n]}$. We may now insert knots using Böhms method as before, which yields the extended knot vector Ξ' and the new control points $\{\boldsymbol{Q}'_i\}_{i\in[1,n+s]} = \{(x'_i, y'_i, w'_i)\}_{i\in[1,n+s]}$. Projecting this B-spline curve back to a 2D NURBS with control points given by

$$\{\boldsymbol{P}'_i\}_{i\in[1,n+s]} = \left\{\left(\frac{x'_i}{w'_i}, \frac{y'_i}{w'_i}\right)\right\}_{i\in[1,n+s]}$$

and weights given by $\{w'_i\}_{i\in[1,n+s]}$, we get the refined NURBS. To insert knots in a 3D NURBS curve, we apply an analogous procedure to the control points $\{\boldsymbol{P}_i\}_{i\in[1,n]} = \{(x_i, y_i, z_i)\}_{i\in[1,n]}$.



(a) NURBS curve with knot locations after refinement.

(b) NURBS curve and control polygon after refinement.

Figure 4.12: Knot insertion for a NURBS circle.

In Figure 4.12 we have inserted the knots $\{0.5, 0.5, 1.5, 2.25, 2.5, 2.75\}$ into the NURBS circle in Figure 4.11. Note that many of the same properties of B-spline curves are preserved for the NURBS curve. Also note that the mesh is not changed by adding an extra knot at $\xi = 0.5$, but we still have added a new basis function. A reduction of continuity in the basis functions will here occur, but the continuity of the curve remains the same.

4.2.2 NURBS volumes

As for spline volumes the extension to trivariate NURBS volumes is also straight forward. Let $\{N_{i,p}\}_{i\in[1,n]}$, $\{M_{j,q}\}_{j\in[1,m]}$ and $\{L_{k,p}\}_{k\in[1,l]}$ be the set of B-spline basis functions in ξ-, η- and ζ-direction, respectively. These set's have their own degree (p, q and r, respectively) and knot vectors (Ξ, $\mathcal{H}$ and $\mathcal{Z}$, respectively). The trivariate NURBS basis functions are then defined by

$$R^{p,q,r}_{i,j,k}(\xi,\eta,\zeta) = \frac{N_{i,p}(\xi)M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k}}{W(\xi,\eta,\zeta)} \tag{4.9}$$

where the weighting function is now given by

$$W(\xi,\eta,\zeta) = \sum_{\hat{\imath}=1}^{n}\sum_{\hat{\jmath}=1}^{m}\sum_{\hat{k}=1}^{l} N_{\hat{\imath},p}(\xi)M_{\hat{\jmath},q}(\eta)L_{\hat{k},r}(\zeta)w_{\hat{\imath},\hat{\jmath},\hat{k}}.$$

The partial derivatives of these functions are then given by the quotient rule. For example, we have

$$\frac{\partial R^{p,q,r}_{i,j,k}(\xi,\eta,\zeta)}{\partial\xi} = \frac{W(\xi,\eta,\zeta)N'_{i,p}(\xi) - W_\xi(\xi,\eta,\zeta)N_{i,p}(\xi)}{(W(\xi,\eta,\zeta))^2}M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k} \tag{4.10}$$

where

$$W_\xi(\xi,\eta,\zeta) = \sum_{\hat{\imath}=1}^{n}\sum_{\hat{\jmath}=1}^{m}\sum_{\hat{k}=1}^{l} N'_{\hat{\imath},p}(\xi)M_{\hat{\jmath},q}(\eta)L_{\hat{k},r}(\zeta)w_{\hat{\imath},\hat{\jmath},\hat{k}}$$

and

$$N'_{i,p}(\xi) = \frac{\mathrm{d}N_{i,p}(\xi)}{\mathrm{d}\xi}.$$

Analogous expressions may be found for the partial derivatives with respect to η and ζ.

A 3D NURBS volume is now defined by

$$\boldsymbol{V}(\xi,\eta,\zeta) = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l} R_{i,j,k}^{p,q,r}(\xi,\eta,\zeta)\boldsymbol{P}_{i,j,k}.$$

Knot insertion into such an object is once again done by inserting knots into a 4D B-splines volume with control points

$$\boldsymbol{Q}_{i,j,k} = (w_{i,j,k}x_{i,j,k}, w_{i,j,k}y_{i,j,k}, w_{i,j,k}z_{i,j,k}, w_{i,j,k})$$

for $i \in [1,n]$, $j \in [1,m]$ and $k \in [1,l]$. Using the same procedure as in Subsection 4.1.3 we obtain the new set of control points $\boldsymbol{Q}'_{i,j,k} = (x'_{i,j,k}, y'_{i,j,k}, z'_{i,j,k}, w_{i,j,k})$ which after the projection yields our new set of control points for the NURBS volume given by

$$\left\{\boldsymbol{P}'_{i,j,k}\right\}_{i,j,k} = \left\{\left(\frac{x'_{i,j,k}}{w'_{i,j,k}}, \frac{y'_{i,j,k}}{w'_{i,j,k}}, \frac{z'_{i,j,k}}{w'_{i,j,k}}\right)\right\}_{i,j,k}$$

and a set of weights given by $\{w'_{i,j,k}\}_{i,j,k}$.

A corresponding procedure is done for degree elevation in NURBS volumes.

4.3 The weak form and Galerkin's method

The *weak form* of the problem is derived from the strong form. Typically, one defines two classes of functions: $\mathcal{S}_i$ denotes the *solution space* and $\mathcal{V}_i$ denotes the *weighting space* for a given spatial component i. These spaces are made in order to handle non homogeneous Dirichlet boundary conditions, but as we shall only consider homogeneous Dirichlet boundary conditions, these spaces will be the same. That is, $\mathcal{S}_i = \mathcal{V}_i$ for $i = 1,2,3$. Typically, $\mathcal{S}_i$ is a subspace of the Sobolev space $H^1(\Omega)$ (which consist of all functions which have square-integrable derivatives) with a condition such that the Dirichlet boundary condition is satisfied.

As an introduction to isogeometric analysis we shall first consider the static linear elasticity case. That is we let the right hand side of (2.12) be zero (no acceleration). Multiply now each of the equations in (2.12) by a corresponding test functions $v_i \in \mathcal{S}_i$ and sum these three equations into one single equation given by

$$v_i\sigma_{ij,j} = -v_i f_i$$

which can be written as

$$v_i \nabla \cdot \bar{\boldsymbol{\sigma}}_i = -v_i f_i$$

where

$$\bar{\boldsymbol{\sigma}}_i = \begin{bmatrix}\sigma_{i1}\\ \sigma_{i2}\\ \sigma_{i3}\end{bmatrix}.$$

Integrating over the domain yields

$$\int_\Omega v_i \nabla \cdot \bar{\boldsymbol{\sigma}}_i \, \mathrm{d}\Omega = -\int_\Omega v_i f_i \, \mathrm{d}\Omega. \tag{4.11}$$

Using formula Equation (A.3) we have

$$\int_\Omega \nabla v_i \cdot \bar{\boldsymbol{\sigma}}_i \, \mathrm{d}\Omega = \int_{\Gamma_{N_i}} v_i \bar{\boldsymbol{\sigma}}_i \cdot \boldsymbol{n} \, \mathrm{d}\Omega + \int_\Omega v_i f_i \, \mathrm{d}\Omega. \tag{4.12}$$

Note that we only integrate over the Neumann part of the boundary since the integral over the Dirichlet part of the boundary vanishes (the test function v_i is zero at this boundary). Also note that the domain of integration for the boundary integral is depending on i. Returning to the index summation convention we may rewrite Equation (4.12) as

$$\int_\Omega v_{i,j}\sigma_{ij} \, \mathrm{d}\Omega = \int_{\Gamma_{N_i}} v_i(\sigma_{ij} n_j) \, \mathrm{d}\Gamma + \int_\Omega v_i f_i \, \mathrm{d}\Omega \tag{4.13}$$

which using the boundary conditions, may be written as

$$\int_\Omega v_{(i,j)}\sigma_{ij} \, \mathrm{d}\Omega = \int_{\Gamma_{N_i}} v_i h_j \, \mathrm{d}\Gamma + \int_\Omega v_i f_i \, \mathrm{d}\Omega.$$

Note that since $[\sigma_{ij}]$ is a symmetric tensor, we have used Equation (2.11) to only write the symmetric part of $v_{i,j}$. If we now define the space $\boldsymbol{\mathcal{S}} = \{\boldsymbol{u} \,|\, u_i \in \mathcal{S}_i\}$ we can state the weak formulation in a concise form: Find $\boldsymbol{u} \in \boldsymbol{\mathcal{S}}$ such that for all $\boldsymbol{v} \in \boldsymbol{\mathcal{S}}$ we have

$$a(\boldsymbol{v}, \boldsymbol{u}) = L(\boldsymbol{v})$$

where

$$a(\boldsymbol{v}, \boldsymbol{u}) = \int_\Omega v_{(i,j)} c_{ijkl} u_{(k,l)} \, \mathrm{d}\Omega \tag{4.14}$$

and

$$L(\boldsymbol{v}) = \int_{\Gamma_{N_i}} v_i h_j \, \mathrm{d}\Gamma + \int_\Omega v_i f_i \, \mathrm{d}\Omega.$$

Here we have used the relation $\sigma_{ij} = c_{ijkl}\varepsilon_{kl} = c_{ijkl}u_{(k,l)}$.

We now want to transform this weak statement into a system of algebraic equations. We here apply Galerkin's method and now turn to a finite-dimensional subspace $\boldsymbol{\mathcal{S}}^h \subset \boldsymbol{\mathcal{S}}$. The basis for this subspace is the presented NURBS basis. But note that we will have vector valued control variables. The Galerkin approximation of the weak form is now given by: Find $\boldsymbol{u}^h \in \boldsymbol{\mathcal{S}}^h$ such that

$$a(\boldsymbol{v}^h, \boldsymbol{u}^h) = L(\boldsymbol{v}^h) \tag{4.15}$$

for all $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h$.

To find the system of algebraic equations we need to write $\boldsymbol{u}^h$ as a linear combination of the basis functions. First, let $\boldsymbol{\eta} = \{1, \ldots, n_{np}\}$ (where n_{np} is the number of basis functions) be the set containing the indices of all the functions in the NURBS basis defining the geometry and let $\boldsymbol{\eta}_{g_i} \subset \boldsymbol{\eta}$ be the set containing the indices of all the basis

functions that are non-zero on Γ_D. Due to the homogeneous Dirichlet boundary condition, we may write the i^{th} component of $\boldsymbol{v}^h$ and the j^{th} component of $\boldsymbol{u}^h$ as

$$v_i^h = \sum_{A \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} R_A c_{iA} \quad \text{and} \quad u_j^h = \sum_{B \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_j}} R_B d_{jB} \tag{4.16}$$

respectively, where $\boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}$ denotes set subtraction. Using the index summation convention we may now write

$$\boldsymbol{v}^h = v_i^h \boldsymbol{e}_i \quad \text{and} \quad \boldsymbol{u}^h = u_j^h \boldsymbol{e}_j \tag{4.17}$$

where the unit vectors $\boldsymbol{e}_i$ are given by

$$\boldsymbol{e}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{e}_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad \text{and} \quad \boldsymbol{e}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The finale step is now to insert Equation (4.17) (using Equation (4.16)) into Equation (4.15) such that we obtain a matrix formulation of the problem. Insertion yields

$$a\left(\sum_{A \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_j}} R_A c_{iA} \boldsymbol{e}_i, \sum_{B \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} R_B d_{jB} \boldsymbol{e}_j \right) - L\left(\sum_{A \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} R_A c_{iA} \boldsymbol{e}_i \right) = 0$$

which using the bilinearity of a and the linearity of L may be written as

$$\sum_{A \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} c_{iA} \left(\sum_{B \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} a\left(R_A \boldsymbol{e}_i, R_B \boldsymbol{e}_j\right) d_{jB} - L\left(R_A \boldsymbol{e}_i\right) \right) = 0.$$

Since the coefficients c_{iA} is arbitrary (the relation should hold for all $\boldsymbol{v}^h \in \boldsymbol{\mathcal{S}}^h$) the term in the parentheses must vanish. That is, for all $A \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}$ and $i = 1, 2, 3$ we have

$$\sum_{B \in \boldsymbol{\eta} - \boldsymbol{\eta}_{g_i}} a\left(R_A \boldsymbol{e}_i, R_B \boldsymbol{e}_j\right) d_{jB} = L\left(R_A \boldsymbol{e}_i\right).$$

One should typically make a system of the ordering of these equations. That is, one should create a function ID which collapse the indices i and A into a single index. A given equation then has the index $P = \text{ID}(i, A)$ and the index over all unknown components of the displacement vectors are called $Q = \text{ID}(j, B)$. The resulting system of equation may then be written as

$$\boldsymbol{K}\boldsymbol{U} = \boldsymbol{F}$$

where

$$\begin{aligned} \boldsymbol{K} &= [K_{PQ}], \\ \boldsymbol{U} &= \{d_Q\}, \\ \boldsymbol{F} &= \{F_P\}, \end{aligned}$$

and

$$\begin{aligned} K_{PQ} &= a\left(R_A \boldsymbol{e}_i, R_B \boldsymbol{e}_j\right), \\ F_P &= L(R_A \boldsymbol{e}_j), \\ d_Q &= d_{jB}. \end{aligned}$$

The matrix $\boldsymbol{K}$ is the so-called *stiffness matrix*.

In the implementations, we shall use the indexing introduces in [17] rather than the indexing in [2], mainly because the first is arguably more clean. The most important difference is the bandwidth, but MATLAB has efficient algorithms to reduce the bandwidth to a minimal in any case, so it poses no problems anyways. Typically, a section about this indexing should be presented, but we shall for now only refer to [17] for details. Later, when considering the fluid-structure interaction problem, we shall create an example to illustrate the main idea.

4.4 Assembly

As for the finite element method, one typically do not loop through the basis functions. Rather, we loop through the elements constructing local stiffness matrices and successfully place their contribution in the global stiffness matrix. Let us first introduce some notations. The elastic coefficients are typically inserted in a matrix $\boldsymbol{C}$ called the elasticity matrix. It is defined by

$$\boldsymbol{C} = \begin{bmatrix} c_{1111} & c_{1122} & c_{1133} & c_{1123} & c_{1113} & c_{1112} \\ & c_{2222} & c_{2233} & c_{2223} & c_{2213} & c_{2212} \\ & & c_{3333} & c_{3323} & c_{3313} & c_{3312} \\ & & & c_{2323} & c_{2323} & c_{2312} \\ & \text{symmetric} & & & c_{1313} & c_{1312} \\ & & & & & c_{1212} \end{bmatrix},$$

or in our case, more explicitly by

$$\boldsymbol{C} = \begin{bmatrix} 2\mu+\lambda & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & 2\mu+\lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & 2\mu+\lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix}.$$

Moreover, using *Voigt notation*, we define the *strain vector* and the *stress vector* to be

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \begin{bmatrix} u_{1,1} \\ u_{2,2} \\ u_{3,3} \\ u_{2,3} + u_{3,2} \\ u_{3,1} + u_{1,3} \\ u_{1,2} + u_{2,1} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{13} \\ \sigma_{12} \end{bmatrix}.$$

Then

$$\boldsymbol{\sigma} = \boldsymbol{C}\boldsymbol{\varepsilon}(\boldsymbol{u}),$$

such that we may write the bilinear form in Equation (4.14) as

$$a(\boldsymbol{v}, \boldsymbol{u}) = \int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{v})^\top \boldsymbol{C}\boldsymbol{\varepsilon}(\boldsymbol{u}) \, \mathrm{d}\Omega.$$

Also note that

$$\boldsymbol{\varepsilon}(R_A \boldsymbol{e}_i) = \boldsymbol{B}_A \boldsymbol{e}_i,$$

where

$$\boldsymbol{B}_A = \begin{bmatrix} R_{A,1} & 0 & 0 \\ 0 & R_{A,2} & 0 \\ 0 & 0 & R_{A,3} \\ 0 & R_{A,3} & R_{A,2} \\ R_{A,3} & 0 & R_{A,1} \\ R_{A,2} & R_{A,1} & 0 \end{bmatrix}. \tag{4.18}$$

The entries in the global stiffness matrix may then be written as

$$K_{PQ} = a\left(R_A \boldsymbol{e}_i, R_B \boldsymbol{e}_j\right) = \boldsymbol{e}_i^\top \int_\Omega \boldsymbol{B}_A^\top \boldsymbol{C} \boldsymbol{B}_B \, \mathrm{d}\Omega \, \boldsymbol{e}_j.$$

Let Ω^e be the domain of a given element, where the index e loops over all elements. The support of the NURBS functions are highly localized. To reduce computations, we should only integrate over functions which have support in Ω^e. If we have n_{en} such *local shape functions*, and let a and b iterate over these functions, we may calculate the entries in the local stiffness matrix as

$$k_{pq}^e = \boldsymbol{e}_i^\top \int_{\Omega^e} \boldsymbol{B}_a^\top \boldsymbol{C} \boldsymbol{B}_b \, \mathrm{d}\Omega \, \boldsymbol{e}_j$$

where

$$p = n_{en}(i-1) + a \quad \text{and} \quad q = n_{en}(j-1) + b.$$

The local force vector may similarly be calculated by

$$f_p^e = \int_{\Gamma_{N_i}^e} R_A h_i \, \mathrm{d}\Gamma + \int_{\Omega_e} R_A f_i \, \mathrm{d}\Omega. \tag{4.19}$$

The integration is done by quadrature formulas. One first maps to the parametric domain, and then map this domain to a parent domain. The element in the parametric domain, corresponding to Ω^e, is given by

$$\hat{\Omega}^e = [\xi_i, \xi_{i+1}] \times [\eta_j, \eta_{j+1}] \times [\zeta_k, \zeta_{k+1}].$$

In three dimensions we want to map this domain into the parent domain given by

$$\tilde{\Omega}^e = [-1, 1] \times [-1, 1] \times [-1, 1].$$

So given $(\tilde{\xi}, \tilde{\eta}, \tilde{\zeta}) \in \tilde{\Omega}^e$, we calculate $(\xi, \eta, \zeta) \in \hat{\Omega}^e$ by

$$\begin{aligned} \xi &= \xi_i + (\tilde{\xi} + 1)\frac{\xi_{i+1} - \xi_i}{2}, \\ \eta &= \eta_j + (\tilde{\eta} + 1)\frac{\eta_{j+1} - \eta_j}{2}, \\ \zeta &= \zeta_k + (\tilde{\zeta} + 1)\frac{\zeta_{k+1} - \zeta_k}{2}. \end{aligned}$$

The Jacobian determinant for the parametric to parent mapping is thus given by

$$J_2 = \begin{vmatrix} \frac{\partial \xi}{\partial \tilde{\xi}} & \frac{\partial \xi}{\partial \tilde{\eta}} & \frac{\partial \xi}{\partial \tilde{\zeta}} \\ \frac{\partial \eta}{\partial \tilde{\xi}} & \frac{\partial \eta}{\partial \tilde{\eta}} & \frac{\partial \eta}{\partial \tilde{\zeta}} \\ \frac{\partial \zeta}{\partial \tilde{\xi}} & \frac{\partial \zeta}{\partial \tilde{\eta}} & \frac{\partial \zeta}{\partial \tilde{\zeta}} \end{vmatrix} = \frac{1}{8}(\xi_{i+1} - \xi_i)(\eta_{j+1} - \eta_j)(\zeta_{k+1} - \zeta_k).$$

Similarly, we need the Jacobian for the mapping from the physical domain into the parametric domain. Given $(\xi, \eta, \zeta) \in \hat{\Omega}^e$, we calculate $(x, y, z) = (x_1, x_2, x_3) \in \Omega^e$ by

$$\boldsymbol{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{k=1}^{l} R_{i,j,k}^{p,q,r}(\xi, \eta, \zeta) \boldsymbol{P}_{i,j,k} = \sum_{a=1}^{n_{en}} R_a(\xi, \eta, \zeta) \boldsymbol{P}_a$$

where $\boldsymbol{P}_{i,j,k}$ are the control points and $R_{i,j,k}^{p,q,r}(\xi, \eta, \zeta)$ are the NURBS basis functions which are computed by Equation (4.9). We shall denote this transformation by $\boldsymbol{X} : \hat{\Omega} \to \Omega$. Note that the last equality again comes from the highly localized support of the NURBS basis. The Jacobian matrix is thus given by

$$\boldsymbol{J} = \begin{bmatrix} \frac{\partial x_1}{\partial \xi} & \frac{\partial x_1}{\partial \eta} & \frac{\partial x_1}{\partial \zeta} \\ \frac{\partial x_2}{\partial \xi} & \frac{\partial x_2}{\partial \eta} & \frac{\partial x_2}{\partial \zeta} \\ \frac{\partial x_3}{\partial \xi} & \frac{\partial x_3}{\partial \eta} & \frac{\partial x_3}{\partial \zeta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}_1, & \boldsymbol{P}_2, & \cdots, & \boldsymbol{P}_{n_{en}} \end{bmatrix} \begin{bmatrix} \frac{\partial R_1}{\partial \xi} & \frac{\partial R_1}{\partial \eta} & \frac{\partial R_1}{\partial \zeta} \\ \frac{\partial R_2}{\partial \xi} & \frac{\partial R_2}{\partial \eta} & \frac{\partial R_2}{\partial \zeta} \\ \vdots & \vdots & \vdots \\ \frac{\partial R_{nen}}{\partial \xi} & \frac{\partial R_{nen}}{\partial \eta} & \frac{\partial R_{nen}}{\partial \zeta} \end{bmatrix} \tag{4.20}$$

such that the Jacobian determinant of this transformation is given by

$$J_1 = \det(\boldsymbol{J})$$

where the derivatives of the NURBS basis functions are computed by Equation (4.10). The matrix $\boldsymbol{B}_a$ contains derivatives of the NURBS functions w.r.t. physical coordinates. So we need to find expressions for $\frac{\partial R}{\partial x_i}$. By the chain rule we have

$$\begin{aligned} \frac{\partial R}{\partial \xi} &= \frac{\partial R}{\partial x_1}\frac{\partial x_1}{\partial \xi} + \frac{\partial R}{\partial x_2}\frac{\partial x_2}{\partial \xi} + \frac{\partial R}{\partial x_3}\frac{\partial x_3}{\partial \xi} \\ \frac{\partial R}{\partial \eta} &= \frac{\partial R}{\partial x_1}\frac{\partial x_1}{\partial \eta} + \frac{\partial R}{\partial x_2}\frac{\partial x_2}{\partial \eta} + \frac{\partial R}{\partial x_3}\frac{\partial x_3}{\partial \eta} \\ \frac{\partial R}{\partial \zeta} &= \frac{\partial R}{\partial x_1}\frac{\partial x_1}{\partial \zeta} + \frac{\partial R}{\partial x_2}\frac{\partial x_2}{\partial \zeta} + \frac{\partial R}{\partial x_3}\frac{\partial x_3}{\partial \zeta}. \end{aligned}$$

And thus, we may write

$$\begin{bmatrix} \frac{\partial R}{\partial x_1}, & \frac{\partial R}{\partial x_2}, & \frac{\partial R}{\partial x_3} \end{bmatrix} \boldsymbol{J} = \begin{bmatrix} \frac{\partial R}{\partial \xi}, & \frac{\partial R}{\partial \eta}, & \frac{\partial R}{\partial \zeta} \end{bmatrix}. \tag{4.21}$$

Multiplying with the inverse of the Jacobian, $\boldsymbol{J}^{-1}$, from the right, and taking the transpose on each side of the equation finally yields

$$\begin{bmatrix} \frac{\partial R}{\partial x_1} \\ \frac{\partial R}{\partial x_2} \\ \frac{\partial R}{\partial x_3} \end{bmatrix} = \boldsymbol{J}^{-\top} \begin{bmatrix} \frac{\partial R}{\partial \xi} \\ \frac{\partial R}{\partial \eta} \\ \frac{\partial R}{\partial \zeta} \end{bmatrix}. \tag{4.22}$$

By successfully placing these expressions in the matrix $\boldsymbol{B}$, we may finally write

$$k_{pq}^e = \boldsymbol{e}_i^\top \int_{\tilde{\Omega}^e} \boldsymbol{B}_a^\top \boldsymbol{C} \boldsymbol{B}_b |J_1||J_2| \, \mathrm{d}\tilde{\Omega} \, \boldsymbol{e}_j.$$

By carefully sequentially placing all values for $a, b = 1, \ldots, n_{en}$ into the matrix $\boldsymbol{B}$ we can compute the whole local stiffness matrix in one go by

$$\boldsymbol{k}^e = \int_{\tilde{\Omega}^e} \boldsymbol{B}^\top \boldsymbol{C} \boldsymbol{B} |J_1||J_2| \, \mathrm{d}\tilde{\Omega}.$$

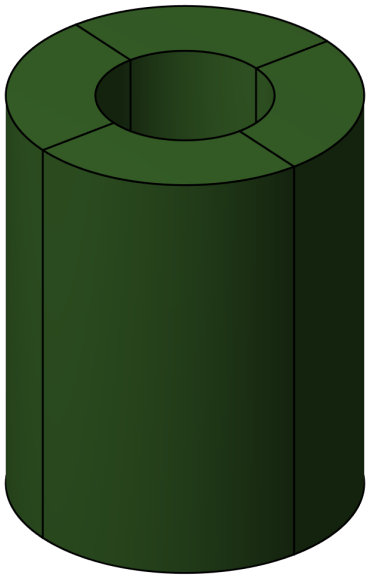


Figure 4.13: Cylinder with 4 elements.

The integrals are approximated with quadrature rules. If we want to approximate the integral

$$\int_{\tilde{\Omega}} g(\tilde{\xi}, \tilde{\eta}, \tilde{\zeta}) \, \mathrm{d}\tilde{\Omega},$$

the approximation by Gaussian quadrature is given by

$$\int_{\tilde{\Omega}} g(\tilde{\xi}, \tilde{\eta}, \tilde{\zeta}) \, \mathrm{d}\tilde{\Omega} \approx \sum_{q=1}^{n_\mathrm{q}} \rho_q g(\tilde{\xi}_q, \tilde{\eta}_q, \tilde{\zeta}_q),$$

where n_q are the number of integration points, and $(\tilde{\xi}_q, \tilde{\eta}_q, \tilde{\zeta}_q)$ and ρ_q are given quadrature points and weights, respectively.

A similar procedure as shown above is needed to approximate the integrals in Equation (4.19). The full expression will depend on which part of the surface we integrate over.

As an example, consider the cylinder in Figure 4.13 where a constant (with value p_o) normal traction is supposed to be applied on the outside surface of the cylinder (with no body forces). The NURBS parametrization is such that ξ traverses the circumferential direction, η traverses the length and ζ traverses the thickness. Hence, the tractions are applied at the surface corresponding to $\zeta = 1$. An integral over such a parameterized surface is evaluated by (cf. [18])

$$\int_\Omega f \, \mathrm{d}S = \int_{\hat{\Omega}} f(\boldsymbol{X}(\xi, \eta, 1)) \left\| \frac{\partial \boldsymbol{X}}{\partial \xi} \times \frac{\partial \boldsymbol{X}}{\partial \eta} \right\| \mathrm{d}\xi \, \mathrm{d}\eta \tag{4.23}$$

where

$$\frac{\partial \boldsymbol{X}}{\partial \xi} = \left[\frac{\partial x_1}{\partial \xi}, \frac{\partial x_2}{\partial \xi}, \frac{\partial x_3}{\partial \xi} \right] \quad \text{and} \quad \frac{\partial \boldsymbol{X}}{\partial \eta} = \left[\frac{\partial x_1}{\partial \eta}, \frac{\partial x_2}{\partial \eta}, \frac{\partial x_3}{\partial \eta} \right].$$

Note that these vectors corresponds to the first and second column of the Jacobi matrix $\boldsymbol{J}$.

As for the local stiffness matrix, we can compute the corresponding contribution to the load vector in one go as follows

$$\boldsymbol{f}^e = \int_{-1}^{1}\int_{-1}^{1} \left[\boldsymbol{R}^\top\Big|_{\zeta=1} h_1, \quad \boldsymbol{R}^\top\Big|_{\zeta=1} h_2, \quad \boldsymbol{R}^\top\Big|_{\zeta=1} h_3\right]^\top \left\|\frac{\partial \boldsymbol{X}}{\partial \xi} \times \frac{\partial \boldsymbol{X}}{\partial \eta}\right\| |J_2| \, \mathrm{d}\tilde{\xi} \, \mathrm{d}\tilde{\eta}$$

where

$$\boldsymbol{R}^\top = \begin{bmatrix} R_1, & R_2, & \ldots, & R_{n_{en}} \end{bmatrix}$$

and

$$J_2 = \frac{1}{4}(\xi_{i+1} - \xi_i)(\eta_{j+1} - \eta_j).$$

Since we consider a normal traction we have

$$h_1 = p_{\mathrm{o}} \cos(\theta), \quad h_2 = p_{\mathrm{o}} \sin(\theta), \quad h_3 = 0.$$

The integral may now be approximated by quadrature rules in two dimensions.

4.5 Post-processing

The dynamic visualizations are done in GLview Inova 9.1, while the static visualizations are done in MATLAB. For the visualizations in GLview, we create so called `.vtf` files directly from MATLAB, and let GLview do the illustrations from here.

We print the nodes, the displacement and every component of the stress for each node into such files. In addition we calculate the von Mises stress given by

$$\sigma_{\mathrm{v}} = \sqrt{\frac{(\sigma_{11} - \sigma_{22})^2 + (\sigma_{22} - \sigma_{33})^2 + (\sigma_{11} - \sigma_{33})^2 + 6(\sigma_{23}^2 + \sigma_{13}^2 + \sigma_{12}^2)}{2}}$$

which we shall use throughout the report in the visualizations.

One must make a grid in the classical FEM style in order to visualize the result. The mesh is simply created by finding the physical coordinates for each physical element. Moreover, for each corner of an element in the parametric space, the displacement and the components of stress can be calculated.

The NURBS mapping often contains singularities, but this is typically avoided as all quadrature points lie on the interior of an element, whereas the singularities typically exist at the boundary of an element. However, in the post-processing step, one evaluates the result at each of the eight corners in each element. If there exist a singularity in the mapping, and the inverse Jacobian must be computed (to find the derivatives) this will then cause problems. In [19] an attempt to solve this problem using symbolic evaluation of the B-spline basis function is presented. However, we shall simply evaluate these points a small distance away from the singularity and use the result of an approximation. After all, the objective is only visualization and not convergence analysis.

The details of the post-processing implementations have been omitted for brevity.

4.6 Error analysis

It is always important to present some numerical evidence that the implementation is correct. This is typically done by finding an analytic solution, and analyze the convergence of the numerical solution towards this analytic solution.

The energy norm is defined by

$$\|\boldsymbol{u}-\boldsymbol{u}^h\|_{\mathrm{E}}=\sqrt{a(\boldsymbol{u}-\boldsymbol{u}^h,\boldsymbol{u}-\boldsymbol{u}^h)}$$

where we compute the bilinear form by

$$\begin{aligned}a(\boldsymbol{u}-\boldsymbol{u}^h,\boldsymbol{u}-\boldsymbol{u}^h)&=\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}-\boldsymbol{u}^h)^\top\boldsymbol{C}\boldsymbol{\varepsilon}(\boldsymbol{u}-\boldsymbol{u}^h)\,\mathrm{d}\Omega\\&=\int_\Omega \left[\boldsymbol{\varepsilon}(\boldsymbol{u})-\boldsymbol{\varepsilon}(\boldsymbol{u}^h)\right]^\top\boldsymbol{C}\left[\boldsymbol{\varepsilon}(\boldsymbol{u})-\boldsymbol{\varepsilon}(\boldsymbol{u}^h)\right]\,\mathrm{d}\Omega\\&=\sum_{e=1}^{n_{el}}\int_{\Omega^e} \left[\boldsymbol{\varepsilon}(\boldsymbol{u})-\boldsymbol{\varepsilon}(\boldsymbol{u}^h)\right]^\top\boldsymbol{C}\left[\boldsymbol{\varepsilon}(\boldsymbol{u})-\boldsymbol{\varepsilon}(\boldsymbol{u}^h)\right]\,\mathrm{d}\Omega,\end{aligned}$$

using the same technique with transformation to the parent element for integration with quadratures. Thus, we need to compute $\varepsilon(\boldsymbol{u})$ in order to do the error analysis.

We shall let the maximal diameter[1] of the elements (in the physical space) be noted by $h=h_{\max}$. We calculate this by looping through all the elements and finding the largest distance between diagonally opposite knot locations.

In [20] we find the following result: Let k be the highest degree of a complete polynomial in the FE basis and denote by $\|\boldsymbol{u}\|_{H^{k+1}}$ the Sobolev norm of order $k+1$. For problems where the solution is not sufficiently smooth, $\boldsymbol{u}\notin H^{k+1}$ we have the error bound

$$\|\boldsymbol{u}-\boldsymbol{u}^h\|_{\mathrm{E}}\leq Ch^\alpha\|\boldsymbol{u}\|_{H^{\alpha+1}}$$

where

$$\alpha=\min\{k,\lambda\}$$

and λ is the strength of the singularity. Such that $\lambda<k$ limits the rate of convergence to the strength of singularity, rather than the polynomial order. This problem may be resolved by using adaptive mesh refinement to obtain $\alpha=k$. That is, on could obtain the same convergence as for sufficiently smooth solutions $\boldsymbol{u}$. The investigation of adaptive mesh refinement (AMR) using T-splines was investigated in [21]. Moreover, the recent RL-Bsplines was introduced in introduced in [22] followed by AMR analysis in [23] and [24].

It has been shown in [7] and [8] that the use of the spline basis functions yields better results. Although the order of convergence remains the same, the constant C is lowered.

The constant C is among other things dependent of the transformation from the parametric space to the physical space. It is not however depending on h. So as h become smaller, we expect the convergence to be of order α. Of course, the analytic solution (and

[1] The maximal diameter of an element is defined to by the diameter of the smallest sphere (in 3D) that can cover the element in the physical space.

then also the norm) should be independent of h, so we shall divide by $\|\boldsymbol{u}\|_{\mathrm{E}}$ to normalize the error. Moreover, when we plot the results, we multiply by 100 to get the normalized error in percentage.

We shall construct two test problems on a solid cylinder. The first is based on the *Solid circular cylinder* problem found in [1], we then construct a more complex example which we name *Kneaded cylinder*. We are then able to plot the convergence in the energy norm. Using degree elevation, we expect the order of convergence to increase correspondingly according to the mentioned result (with $\alpha = k$ as there are no singularities present). Finally, we consider a test problem on a spherical shell.

4.6.1 Solid circular cylinder

Consider a cylinder with inner radius R_{i}, outer radius R_{o} and length L. The *Solid circular cylinder* is subject to an internal constant pressure p_{i} on the inner surface, and a external constant pressure p_{o} on the outer surface. The two ends of the cylinder are fixed in the axial direction z, and are otherwise considered free in the other two spatial dimensions. The analytic solution is found in Appendix C.5 (and in [1][2]) which we repeat here

$$u_{\mathrm{r}}(r) = \frac{1}{2}\frac{1}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\left(\frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{\lambda + \mu} r + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2}{\mu r}(p_{\mathrm{i}} - p_{\mathrm{o}})\right), \tag{4.24}$$

with the following stress field

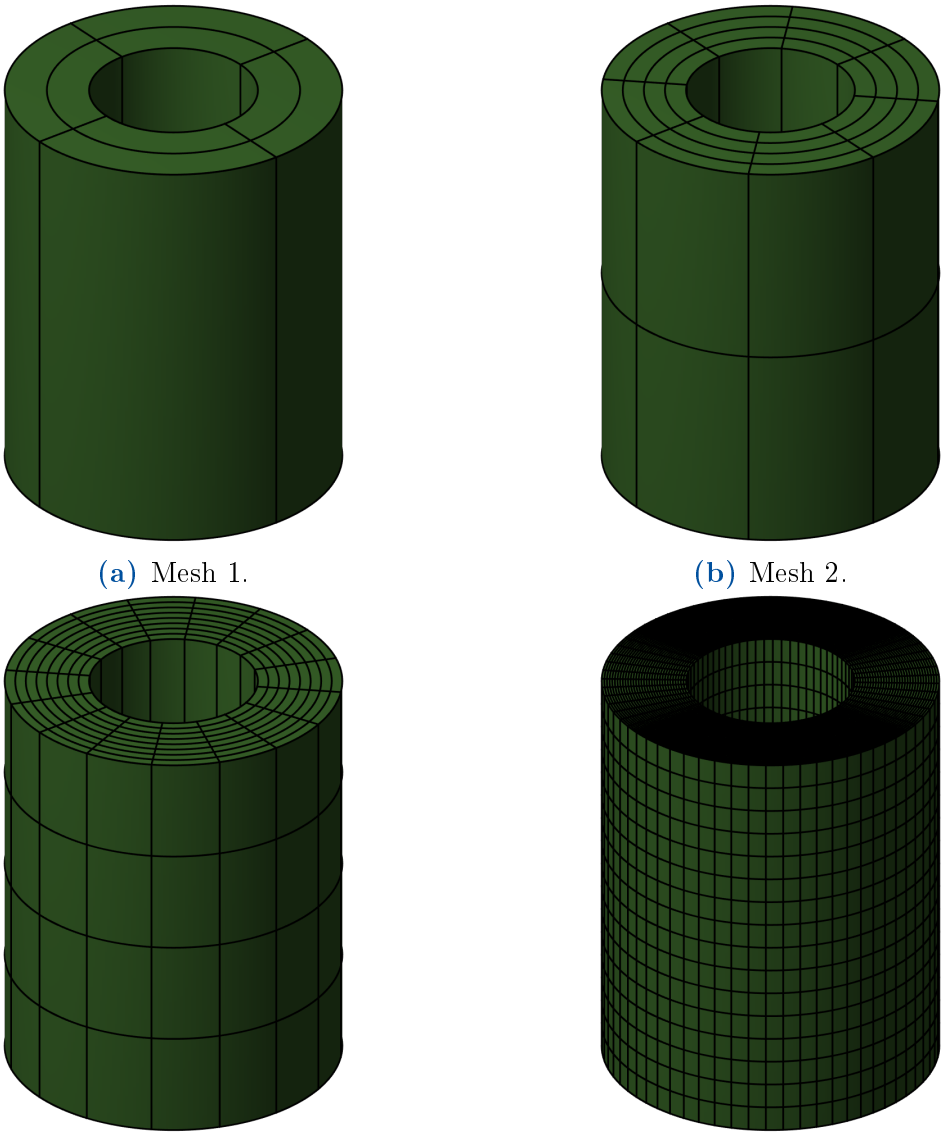
$$\begin{aligned}
\sigma_{\mathrm{rr}} &= \frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{o}} - R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{i}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\frac{1}{r^2} \\
\sigma_{\theta\theta} &= \frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{i}} - R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\frac{1}{r^2} \\
\sigma_{\mathrm{zz}} &= \frac{2\nu(R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}})}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} \\
\sigma_{\theta\mathrm{z}} &= 0 \\
\sigma_{\mathrm{rz}} &= 0 \\
\sigma_{\theta\mathrm{r}} &= 0.
\end{aligned}$$

The knot vector, control points and polynomial orders for the coarsest mesh are tabulated in Appendix D.1. The meshes used in the analysis are found in Figure 4.14 where $R_{\mathrm{o}} = 2$, $R_{\mathrm{i}} = 1$ and $L = 5$. Note that we start by inserting a knot $\zeta = 0.5$ before refining uniformly in all parameter direction. This is done since it is known that the solution only varies radially (that is, in ζ-direction), and we get better numerical result by adding some extra refinement in this direction. Moreover, we use $E = 13$, $\nu = 0.3$, $p_{\mathrm{i}} = 1$ and $p_{\mathrm{o}} = 0$. The solution is shown in Figure 4.15, where axisymmetric response is obtained. The resulting convergence analysis plot is given in Figure 4.16, where we observe the convergence to tend toward the expected order. As the quality of the numerical evidence is reduced with the simplicity of the analytic solution, we shall now present a more complex case.

[2] The original article has a misprint in the analytic solution. The correct solution is

$$u_{\mathrm{r}}(r) = \frac{1}{E}\frac{P R_{\mathrm{i}}^2}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\left((1-2\nu)(1+\nu)r + \frac{R_{\mathrm{o}}^2(1+\nu)}{r}\right)$$

which can be written as Equation (4.24) with $p_{\mathrm{i}} = P$ and $p_{\mathrm{o}} = 0$.



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 5.

Figure 4.14: Solid circular cylinder: Meshes.

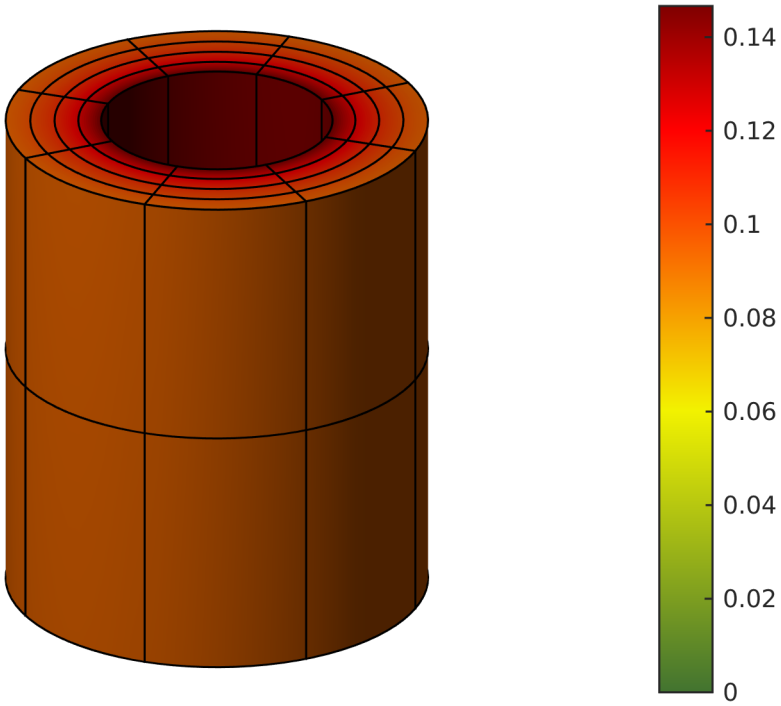


Figure 4.15: Solid circular cylinder: Visualization plot showing purely radial displacement.

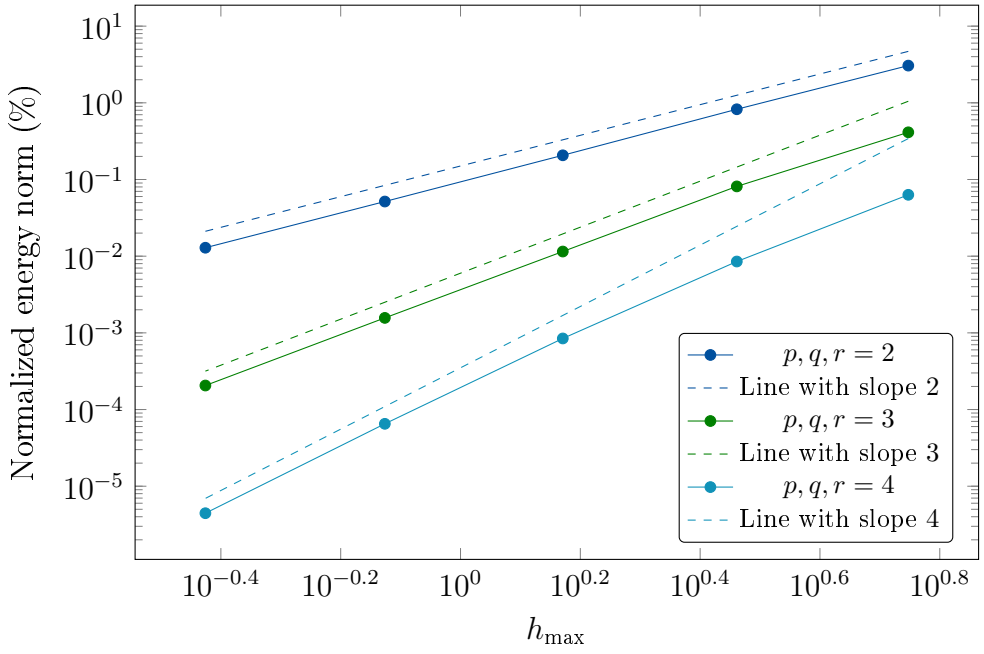


Figure 4.16: Solid circular cylinder: Convergence plot.

4.6.2 Kneaded cylinder

We consider here the same geometry as for the Solid circular cylinder (we also use the same value for the parameters E and ν). To find an analytic solution on this domain, we find a satisfactory solution with respect to some Dirichlet condition and use this to find the body loading and other boundary conditions. Consider the solution

$$\boldsymbol{u} = \frac{L\nu}{\pi\lambda}\begin{bmatrix}\sin\left(\frac{c_1\pi z}{L}\right)\\ \sin\left(\frac{c_2\pi z}{L}\right)\\ \sin\left(\frac{c_3\pi z}{L}\right)\end{bmatrix}$$

where $c_i, i = 1, 2, 3$, are some chosen integers. Obviously this solution satisfies pure homogeneous Dirichlet conditions at both ends of the cylinder (at $z = 0$ and $z = L$).

The stress field is given in cylinder coordinates by (cf. Appendix B.2)

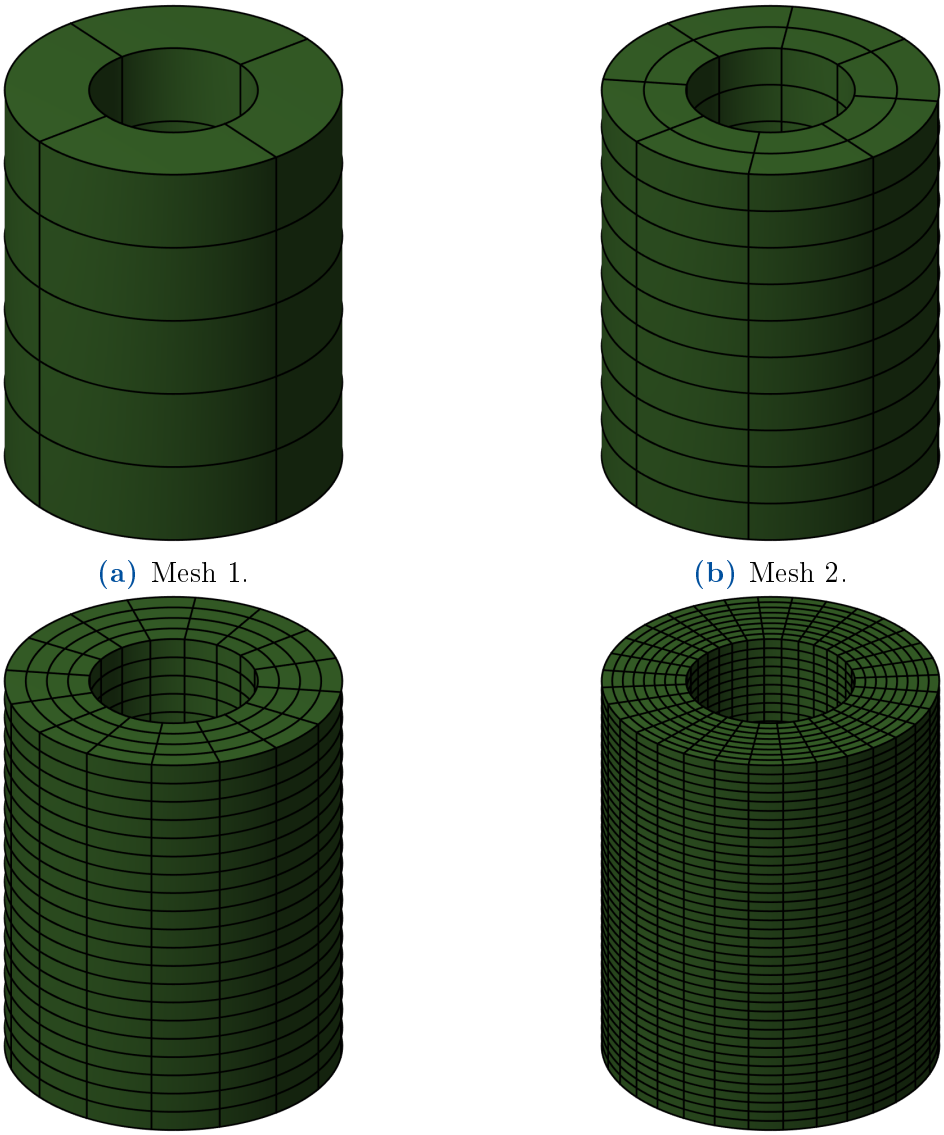
$$\begin{aligned}
\sigma_{\mathrm{rr}} &= \nu c_3 \cos\left(\frac{c_3\pi z}{L}\right)\\
\sigma_{\theta\theta} &= \nu c_3 \cos\left(\frac{c_3\pi z}{L}\right)\\
\sigma_{\mathrm{zz}} &= (1-\nu) c_3 \cos\left(\frac{c_3\pi z}{L}\right)\\
\sigma_{\theta\mathrm{z}} &= \frac{1}{2}(1-2\nu)\left[c_2\cos\left(\frac{c_2\pi z}{L}\right)\cos\theta - c_1\cos\left(\frac{c_1\pi z}{L}\right)\sin\theta\right]\\
\sigma_{\mathrm{rz}} &= \frac{1}{2}(1-2\nu)\left[c_2\cos\left(\frac{c_2\pi z}{L}\right)\sin\theta + c_1\cos\left(\frac{c_1\pi z}{L}\right)\cos\theta\right]\\
\sigma_{\theta\mathrm{r}} &= 0.
\end{aligned}$$

and the body force field is given by

$$\boldsymbol{f} = \frac{\pi}{2L}\begin{bmatrix}(1-2\nu)c_1^2\sin\left(\frac{c_1\pi z}{L}\right)\\ (1-2\nu)c_2^2\sin\left(\frac{c_2\pi z}{L}\right)\\ 2(1-\nu)c_3^2\sin\left(\frac{c_3\pi z}{L}\right)\end{bmatrix}.$$

Both these fields are found using Maple. For the convergence analysis we shall consider two cases. The first case, we set $c_1, c_2, c_3 = 1$ and the second we set $c_1 = 7, c_2 = 5, c_3 = 1$. The meshes used in the analysis are found in Figure 4.17 where $R_{\mathrm{o}} = 2$, $R_{\mathrm{i}} = 1$ and $L = 5$. Note that we start by inserting the knots $\eta = \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}$ before refining uniformly in all parameter direction. This is done since it is known that the solution only varies with z (that is, in η-direction), and we get better numerical result by adding some extra refinement in this direction. Numerical solutions for the cases $c_1, c_2, c_3 = 1$ and $c_1 = 7, c_2 = 5, c_1 = 1$ are shown in Figure 4.18a and Figure 4.18b, respectively.

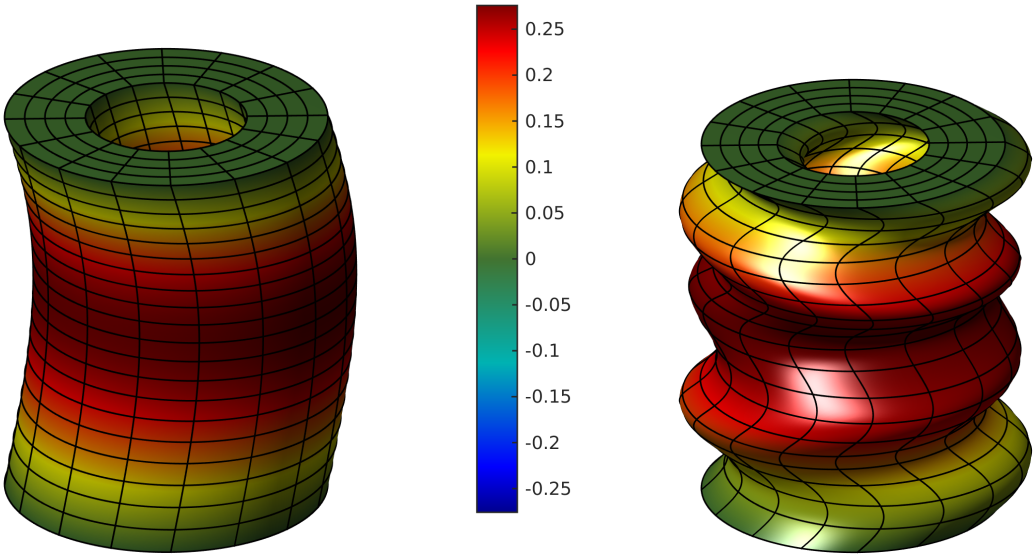
As for the Solid circular cylinder, we get good results for the case $c_1, c_2, c_3 = 1$ as can be seen in Figure 4.19. When we add more oscillation to the solution by increasing values for the parameters c_1, c_2 and c_3, we get a slower start for the convergence, but after some refinements, we eventually also here reproduce the expected convergence rates.



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 4.

Figure 4.17: Kneaded cylinder: Meshes.



(a) $c_1, c_2, c_3 = 1$ on mesh 3.

(b) $c_1 = 7, c_2 = 5, c_3 = 1$ on mesh 3.

Figure 4.18: Kneaded cylinder: Visualization plot of displacement colored by the displacement in the vertical direction (z direction). Both simulations have used $p = q = r = 4$.

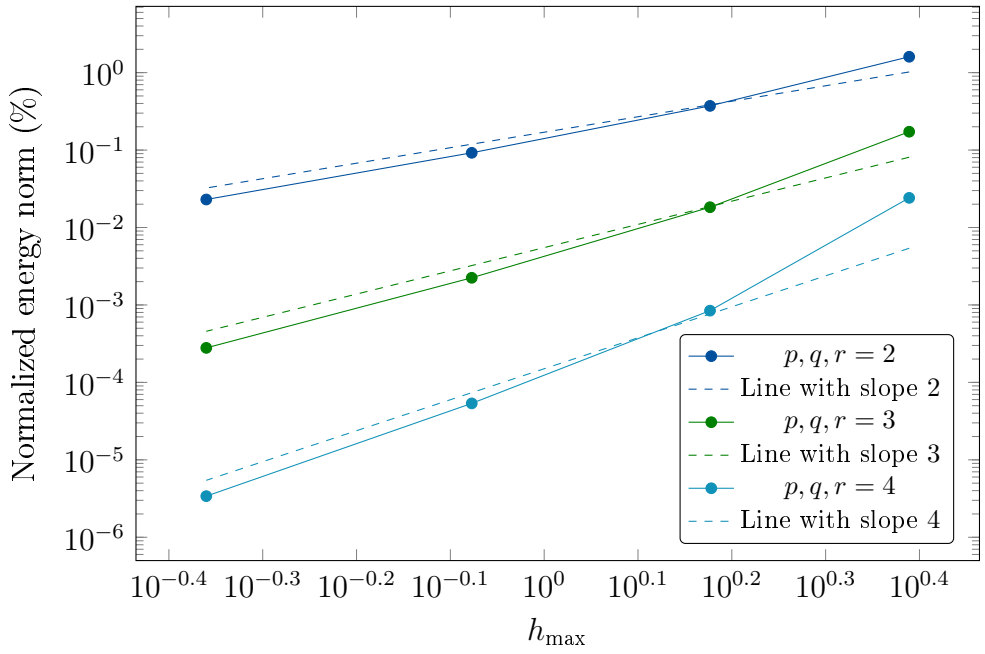


Figure 4.19: Kneaded cylinder: Convergence plot for the case $c_1, c_2, c_3 = 1$.

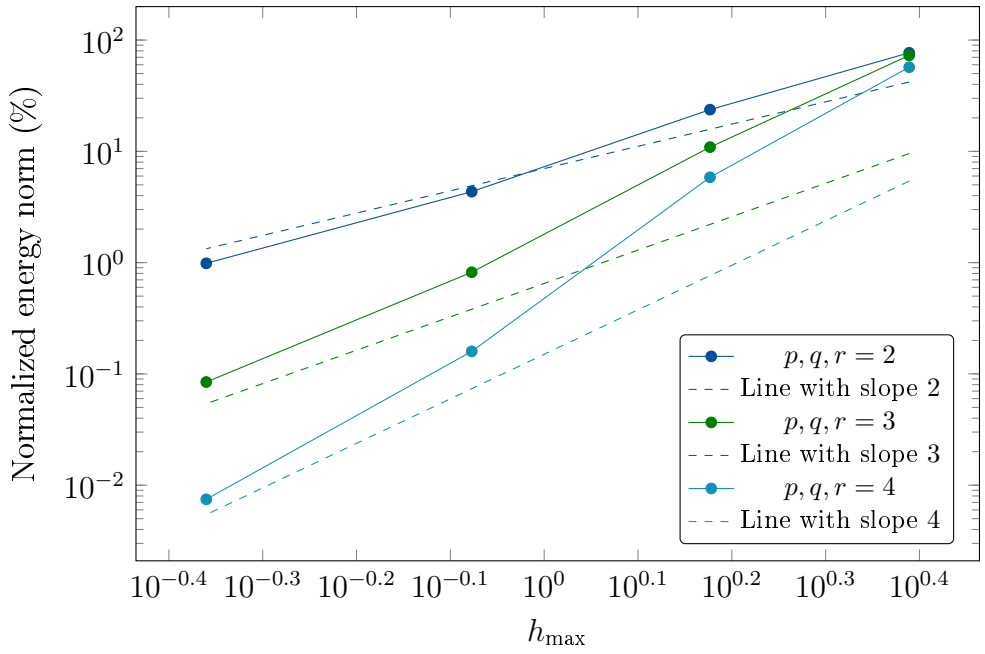


Figure 4.20: Kneaded cylinder: Convergence plot for the case $c_1 = 7, c_2 = 5, c_3 = 1$.

4.6.3 Spherical shell

We shall consider a sphere with inner radius R_i and outer radius R_o. The *spherical shell* is subject to an internal constant pressure p_i on the inner surface, and a external constant pressure p_o on the outer surface. The analytic solution is found in Appendix C.6 which we repeat here

$$u_\mathrm{r}(r) = \frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 p_\mathrm{i} - R_\mathrm{o}^3 p_\mathrm{o}}{3\lambda + 2\mu} r + \frac{1}{4}\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{\mu r^2}\right).$$

with the following stress field (cf. Appendix B.2)

$$\begin{aligned}
\sigma_{\mathrm{rr}} &= \frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{o} - p_\mathrm{i})}{r^3} + R_\mathrm{i}^3 p_\mathrm{i} - R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\theta\theta} &= \frac{1}{2}\frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{r^3} + 2R_\mathrm{i}^3 p_\mathrm{i} - 2R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\phi\phi} &= \frac{1}{2}\frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{r^3} + 2R_\mathrm{i}^3 p_\mathrm{i} - 2R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\theta\phi} &= 0 \\
\sigma_{\mathrm{r}\phi} &= 0 \\
\sigma_{\mathrm{r}\theta} &= 0.
\end{aligned}$$

For the convergence analysis, we shall use the data from Table 3.1 and the pressures are chosen to be $p_\mathrm{i} = 1$ and $p_\mathrm{o} = 2$. The NURBS data for the spherical shell is found in Appendix D.5. Due to the spherical symmetry of the solution we start by some extra refinement in the radial direction. More specifically, we insert the knots $\zeta = 0.25, 0.5, 0.75$. In this section we want to show that it is possible to obtain superconvergence. That is, by using a priori knowledge of the model (symmetry) we may adjust the refinement to be

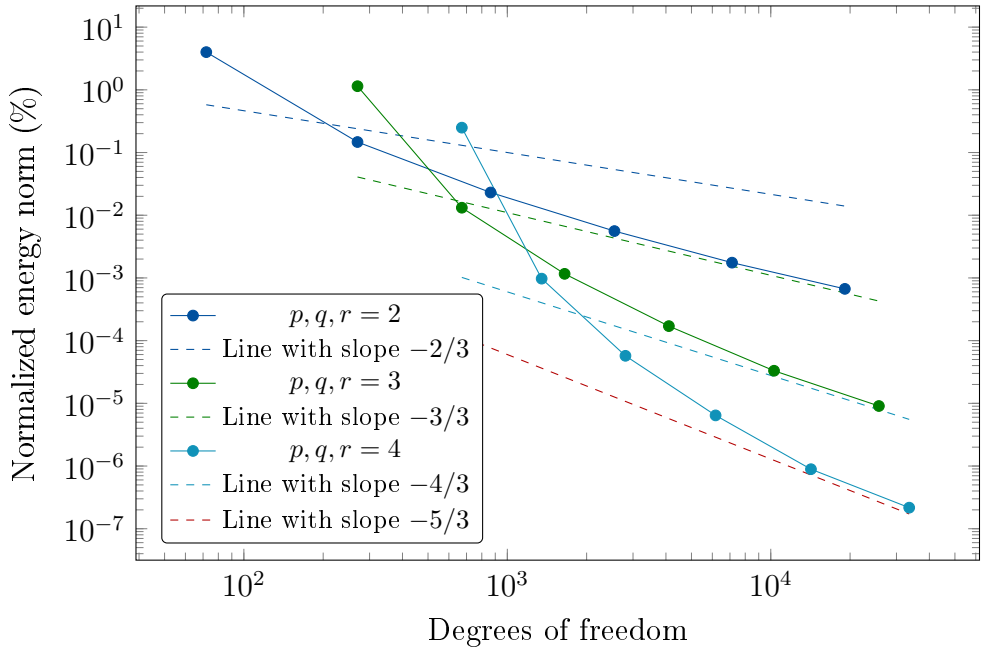


Figure 4.21: Spherical shell: Convergence plot.

non-uniform in all parameter directions. We shall do the refinement process by inserting $2^{s-1} - 1$ in the ζ direction, and $s - 1$ knots in the other directions (here s is the mesh number).

As we no longer use uniform mesh refinement, there makes no more sense to plot the error vs $h_{\max}$. Rather, we plot vs the number of degrees of freedom n_{eq} (the total number of equations which are to be solved). When doing a uniform mesh refinement, one have relation between the mesh parameter $h = h_{\max}$ and n_{eq}, namely (in 3D)

$$h \sim \frac{1}{n_{\text{eq}}^{1/3}}$$

such that the error is bounded by

$$\|\boldsymbol{u} - \boldsymbol{u}^h\|_{\text{E}} \leq C' n_{\text{eq}}^{-k/3}.$$

where C' is just a new constant independent of n_{eq}. For a uniform mesh refinement we thus expect the convergence to be of order $-k/3$ when plotting against the number of degrees of freedom.

The resulting convergence analysis plot is given in Figure 4.21. We observe that the convergence order is raised approximately one whole level (namely from $-k/3$ to $-(k+1)/3$).

Chapter 5

Isogeometric analysis of vibrations

A fundamental mathematical assumption of this thesis, is the harmonic time dependency assumption (see Equation (2.1)). When we transform our time dependent problem to a frequency dependent problem, it is advantageous to analyze the phenomena of vibrations (also named *spectrum analysis*). As it turns out, there are examples where IGA is demonstrably better in this analysis compared to ordinary FEA. That is, one may wonder if the NURBS basis functions (with their smooth properties) serve a better basis than the typical Lagrange basis function which are commonly used in FEM. We shall illustrate this improvement by an example taken from [2]. The work in [25] and [26] investigates the improvement of IGA even further.

5.1 Longitudinal vibrations of an elastic rod

We shall consider an one dimensional rod of length L such that our spatial domain is given by $\Omega = (0, L)$, where we apply homogeneous Dirichlet boundary conditions. That is, $u = 0$ at $\Gamma = \partial\Omega = \{0, L\}$. In one dimension Equation (2.12) reads

$$E\frac{\partial^2 u}{\partial x^2} + \omega^2 \rho u = 0 \tag{5.1}$$

where it is assumed that E and ρ are constant throughout the domain. Proceeding in the usual way, we construct a solution space $\mathcal{S} = \{u | u \in H^1(\Omega), u|_\Gamma = 0\}$. As we have homogeneous Dirichlet boundary conditions, the weighted space $\mathcal{V}$ is identical to the solution space $\mathcal{S}$. We now multiply Equation (5.1) by an arbitrary $v \in \mathcal{V}$ and integrate the resulting equation over the domain.

$$\int_0^L vE\frac{\partial^2 u}{\partial x^2}\,\mathrm{d}x + \omega^2 \int_0^L v\rho u\,\mathrm{d}x = 0$$

Using partial integration we get

$$\left[vE\frac{\partial u}{\partial x}\right]_0^L - \int_0^L \frac{\partial v}{\partial x}E\frac{\partial u}{\partial x}\,\mathrm{d}x + \omega^2 \int_0^L v\rho u\,\mathrm{d}x = 0$$

Exploiting now the fact that $v(0) = v(L) = 0$, the resulting weak form is given by

$$a(v, u) - \omega^2(v, \rho u) = 0$$

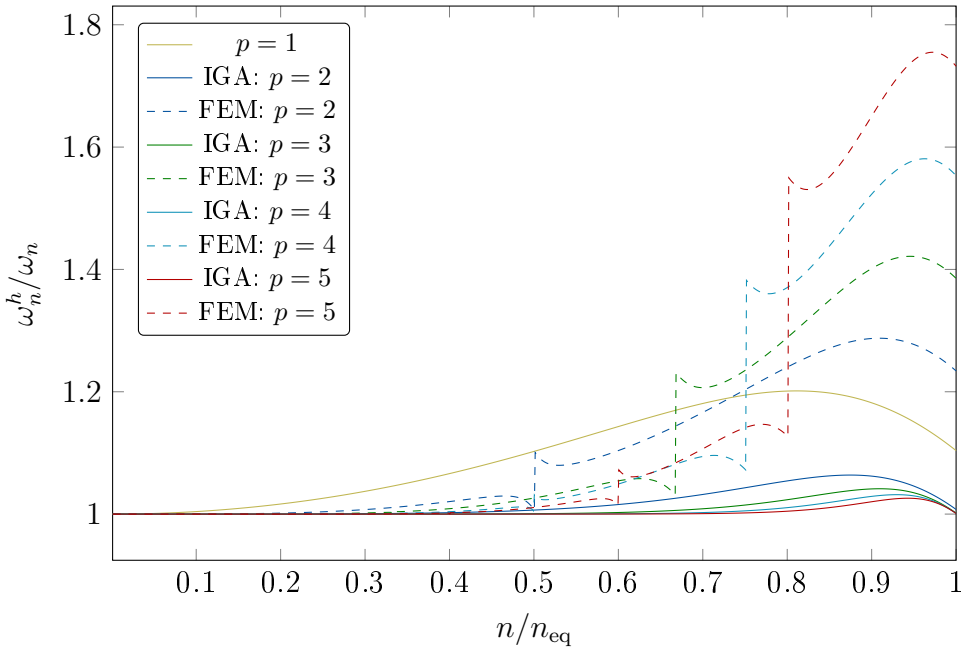


Figure 5.1: Longitudinal vibrations of an elastic rod: Comparison of FEM and IGA. Note that the basis functions for both FEM and IGA are equal in the case $p = 1$. The results are thus identical in this case.

where

$$a(v,u) = \int_0^L \frac{\partial v}{\partial x} E \frac{\partial u}{\partial x}\,\mathrm{d}x$$
$$(v,\rho u) = \omega^2 \int_0^L v\rho u\,\mathrm{d}x$$

Turning now to the Galerkin formulation, we restrict ourselves to solution of the form

$$u^h = \sum_{A=1}^{n_{\text{eq}}} N_A d_A \quad \text{and} \quad u^h = \sum_{B=1}^{n_{\text{eq}}} N_B d_B$$

such that our eigenvalue problem now becomes: Find $\omega^h \in \mathbb{R}^+$ and $u^h \in \mathcal{S}^h$ such that for all $v^h \in \mathcal{V}^h$ we have

$$a(v^h,u^h) - (\omega^h)^2(u^h,\rho v^h) = 0$$

In the analysis we use a nonlinear parametrization of the rod such that the control points are uniformly placed in the physical space. As noted in [2], this result in better results when studying structural vibrations.

Using a finite number of NURBS basis functions[1] (n_{eq}), we will get the same number of eigenvalues ω_n^h.

The solution of the *eigenvalue problem* in Equation (5.1) is trigonometric equations. With the boundary conditions we get the eigenfunctions $u_n(x) = \sin\left(\sqrt{\frac{\rho}{E}}\omega_n x\right)$ and eigenvalues $\omega_n = \sqrt{\frac{E}{\rho}}\frac{n\pi}{L}$. We consider the case $L = 1$, $\rho = 1$, $E = 1$ and $n_{\text{eq}} = 999$. Then the

[1] As all the weights are 1, we actually only get simple B-splines; a subset of NURBS functions.

eigenvalues are simply $\omega_n = n\pi$. We want to compare these analytic eigenvalues to our numerical eigenvalues ω_n^h, by looking at the ratio ω_n/ω_n^h, which should tend to 1 as the numerical solution becomes better. In Figure 5.1 we plot this ratio against the ratio n/n_{eq} using standard Lagrange basis functions (FEM) and B-splines (IGA). A motivation for using IGA in stead of FEM in this thesis is quite apparent in this plot. It should be noted that the B-spline basis functions have larger support and the resulting matrix would then be less sparse. This will in turn increase the computational time for finding the eigenvalues. So the smooth results of IGA comes at some cost in computational time of solving the resulting system of equations. For a more detailed explanation of the different behaviors in Figure 5.1, we refer to Cotterell et al. [2].

5.2 Elastic vibration in 3D

Moving on to three dimensions, we multiply each of the equations in (2.15) by a corresponding test functions $v_i \in \mathcal{S}_i$ and sum these three equations into one single equation given by

$$v_i \sigma_{ij,j} + \omega^2 \rho_{\text{s}} v_i u_i = 0.$$

Note that we have set $f_i = 0$ as we shall not consider any body forces.

Integration over the domain and using the same procedure as for the static linear elasticity case, get the linear form

$$L(\boldsymbol{v}) = \int_{\Gamma_N} v_i h_j \, \mathrm{d}\Gamma.$$

and the bilinear form

$$a(\boldsymbol{v}, \boldsymbol{u}) = \int_\Omega v_{(i,j)} c_{ijkl} u_{(k,l)} \, \mathrm{d}\Omega - \omega^2 \rho_{\text{s}} \int_\Omega v_i u_i \, \mathrm{d}\Omega$$

where the only difference from before is the last term which will result in the *mass matrix* $\boldsymbol{M}$. So if $\boldsymbol{K}$ is the same stiffness matrix as in the static linear elasticity case, we now have the following eigenvalue problem: Find natural frequency $\omega_k^h \in \mathbb{R}^+$ and eigenvectors $\boldsymbol{U}_k$ such that

$$(\boldsymbol{K} - [\omega_k^h]^2 \boldsymbol{M}) \boldsymbol{U}_k = \boldsymbol{0}$$

where (using the same notation as before) the mass matrix is given by

$$\boldsymbol{M} = [M_{PQ}]$$

and

$$M_{PQ} = \rho_{\text{s}} \boldsymbol{e}_i \int_\Omega R_A R_B \, \mathrm{d}\Omega \boldsymbol{e}_j.$$

As $\boldsymbol{e}_i \cdot \boldsymbol{e}_j = \delta_{ij}$ we may write

$$M_{PQ} = \rho_{\text{s}} \delta_{ij} \int_\Omega R_A R_B \, \mathrm{d}\Omega.$$

In the same way the stiffness element matrix could be computed in one go, so can the *element mass matrix*:

$$\boldsymbol{m}^e = \rho_{\text{s}} \int_{\tilde{\Omega}^e} \text{blkdiag}\left(\boldsymbol{R}^\top \boldsymbol{R}, \boldsymbol{R}^\top \boldsymbol{R}, \boldsymbol{R}^\top \boldsymbol{R}\right) |J_1||J_2| \, \mathrm{d}\tilde{\Omega}.$$

Table 5.1: Parameters for circular plate.

Parameter	Description
$E = 30 \cdot 10^9 \text{Pa}$	Young's modulus
$\rho = 2.32 \cdot 10^3 \text{kg/m}^3$	Density
$\nu = 0.2$	Poisson's ratio
$t = 0.02\text{m}$	Thickness of the plate
$R = 2\text{m}$	Radius of plate

where (once again)

$$\boldsymbol{R}^\top = \begin{bmatrix} R_1, & R_2, & \dots, & R_{n_{\text{en}}} \end{bmatrix}$$

and blkdiag() creates a block diagonal matrix with it's arguments on the diagonal.

5.2.1 Circular plate vibrating in vacuum

In [27, pp. 185-189] we find the exact Poisson-Kirchhoff solution to the clamped circular plate. One finds the eigenvalues by first solving

$$\mathrm{i}^{-n} J_n(\mathrm{i}x) J_{n-1}(\mathrm{i}x) - \mathrm{i}^{-n-1} J_n(x) J_{n-1}(\mathrm{i}x) = 0, \quad n = 0, 1, 2, \dots$$

where J_n are the mentioned Bessel functions of the first kind. This equation has infinitely many solution (as for the spherical shell), so we obtain a set of solutions $\{x_{nm}\}$. The eigenvalues are then given by $\beta_{nm} = x_{nm}/R$ such that the natural frequencies may be found by

$$\omega_{nm} = \beta_{nm}^2 \sqrt{\frac{D_E}{\rho t}}, \quad D_E = \frac{E t^3}{12(1-\nu^2)}. \tag{5.2}$$

In [1] the notation $C_{nm} = x_{nm}/\pi$ has been used. In [27, p. 188] there is a misprint in the first eigenfrequency: $C_{01} = 1.015$ where the true value is $C_{01} = 1.0173886$. Hughes et al. copies this misprint in [1] such that he compares his result with $\omega_{01} = 53.863$ when the true value is $\omega_{01} = 54.117$. Hence, in [1] there seems to be convergence to the wrong result when this is not the case. The NURBS data for the coarsest geometry of the plate shell may be found in Appendix D.4, here two parametrizations for the circular plate is presented (cf. Figure 5.2a and Figure 5.2b). In [28] better convergence results was shown for parametrization 2, but we shall restrict ourselves to the analysis on the first of these parametrizations as the main objective is to compare with the results in [2]. Some deviation in the comparison of the results occured, but the main trend is the same.

In Table 5.3 we show the p-convergence of the first 4 computed frequencies. The corresponding modes are shown in Figure 5.3. We can here observe that for axisymmetric modes (ω_{0n}, $n = 0, 1, 2, \dots$) there is hardly any accuracy to be gained by increasing the polynomial order in the angular direction (ξ-direction).

5.2.2 Spherical shell vibrating in vacuum

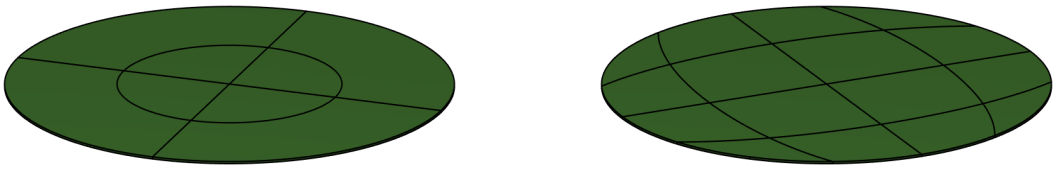
We shall repeat the analysis we did on the circular plate with the spherical shell with data from Table 3.1. The NURBS data for the coarsest geometry of the spherical shell may

Table 5.2: Circular plate: Eigenvalues.

n	ω_{n1}	ω_{n2}	ω_{n3}	ω_{n4}
0	54.117 023	210.682 532	472.017 719	837.960 493
1	112.624 217	322.232 049	636.104 280	1 054.459 619
2	184.756 600	448.065 685	814.815 589	1 285.780 859
3	270.325 036	588.121 621	1 008.109 746	1 531.893 221
4	369.045 760	742.203 564	1 215.845 012	1 792.689 901
5	480.678 086	910.104 164	1 437.855 808	2 068.038 033
6	605.026 114	1 091.631 987	1 673.977 740	2 357.799 001
7	741.929 168	1 286.616 266	1 924.056 001	2 661.837 073
8	891.253 404	1 494.905 872	2 187.947 731	2 980.023 005
9	1 052.885 574	1 716.366 915	2 465.522 082	3 312.235 331
10	1 226.728 502	1 950.880 273	2 756.659 429	3 658.360 594
11	1 412.697 772	2 198.339 383	3 061.250 322	4 018.293 077
12	1 610.719 268	2 458.648 383	3 379.194 403	4 391.934 323
13	1 820.727 309	2 731.720 568	3 710.399 406	4 779.192 571
14	2 042.663 215	3 017.477 105	4 054.780 248	5 179.982 197
15	2 276.474 191	3 315.845 980	4 412.258 234	5 594.223 168
16	2 522.112 431	3 626.761 115	4 782.760 364	6 021.840 543
17	2 779.534 406	3 950.161 633	5 166.218 723	6 462.764 017
18	3 048.700 281	4 285.991 239	5 562.569 959	6 916.927 509
19	3 329.573 436	4 634.197 695	5 971.754 820	7 384.268 790
20	3 622.120 072	4 994.732 375	6 393.717 757	7 864.729 156

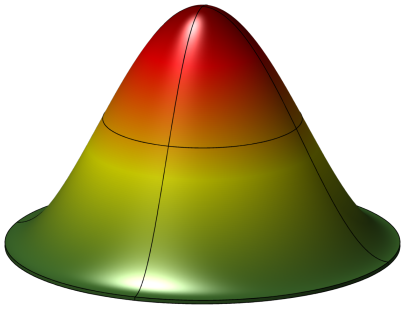
Table 5.3: Circular plate: Convergence of computed eigenfrequencies.

p	q	r	ω_{01}	ω_{11}	ω_{21}	ω_{02}
2	2	2	254.838	778.860	2569.078	3456.036
2	3	2	54.424	138.174	400.934	214.661
3	3	2	54.424	117.358	196.698	214.661
3	4	2	54.253	116.359	187.731	211.796
4	4	2	54.253	112.954	185.972	211.796
4	5	2	54.203	112.801	185.475	210.912
5	5	2	54.203	112.783	185.218	210.912
5	6	2	54.177	112.728	185.122	210.803
exact			54.117	112.624	184.756	210.682

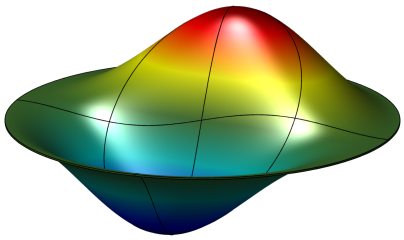


(a) Parametrization 1. **(b)** Parametrization 2.

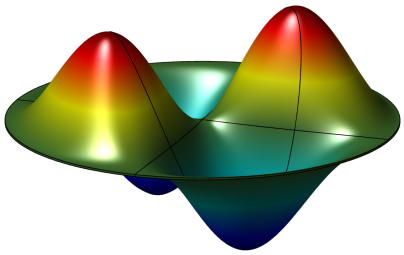
Figure 5.2: Circular plate: Example of parametrizations of the circular plate.



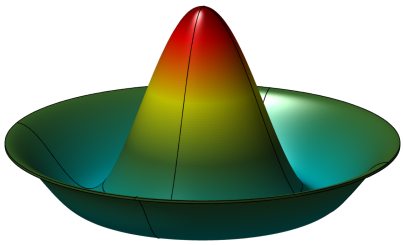
(a) Mode 1.



(b) Mode 2.



(c) Mode 4.



(d) Mode 6.

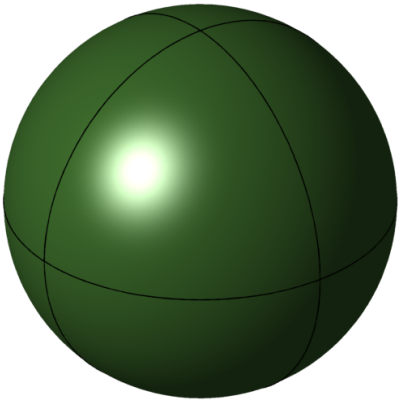
Figure 5.3: Circular plate: Four vibration modes using $p = 6$, $q = 6$ and $r = 2$ with parametrization 1.

Table 5.4: Spherical shell: Convergence of computed eigenfrequencies.

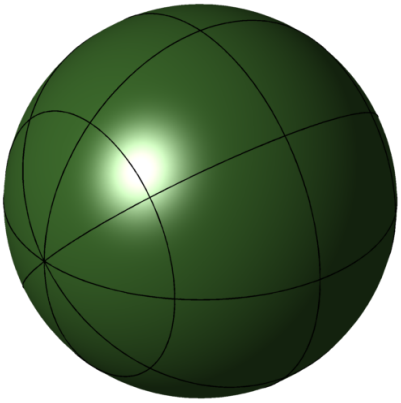
p	q	r	ω_{21}	ω_{31}	ω_{41}	ω_{51}
2	2	2	777.802	1019.834	1379.161	2159.979
3	3	2	764.428	913.290	1000.350	1235.107
4	4	2	764.175	908.693	976.975	1038.282
5	5	2	764.123	908.041	972.766	1021.684
exact			764.121	907.967	972.169	1017.829

be found in Appendix D.5. In this analysis we shall insert the knots $\{\frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}\}$ in the ξ-direction and $\{0.25, 0.75\}$ in the η-direction. We do not refine the mesh in ζ-direction. This results in mesh 2 in Figure 5.4.

The results is shown in Table 5.4 where the exact eigenfrequencies are collected from Table 3.2. As the spherical shell is not clamped, the first 6 modes (3 from translation and 3 from rotation) will correspond to eigenvalue equal zero. The first interesting mode is thus vibration mode 7, which corresponds to ω_{21}. Up to rotation, there is 4 identical other modes (identical eigenvalues), such that the next mode is vibration mode 12 which then corresponds to ω_{31}. The first four unique modes up to rotation is shown in Figure 5.5. We note that the accuracy decreases for modes with higher frequencies.

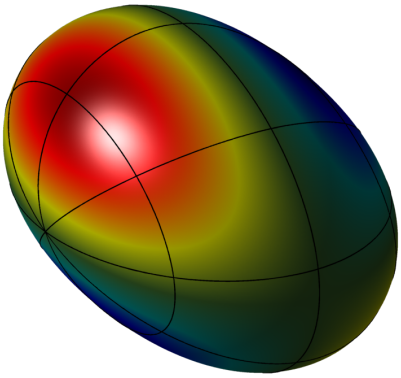


(a) Mesh 1.

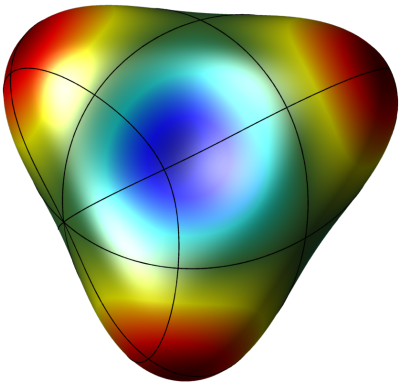


(b) Mesh 2.

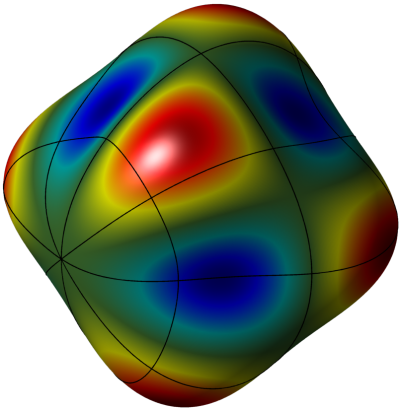
Figure 5.4: Spherical shell: The first two meshes with, respectively, 8 and 32 elements.



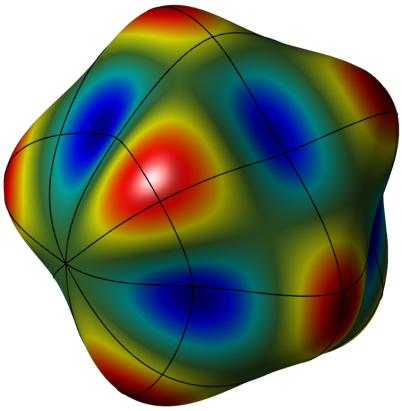
(a) Mode 7 corresponding to ω_{21}.



(b) Mode 12 corresponding to ω_{31}.



(c) Mode 19 corresponding to ω_{41}.



(d) Mode 28 corresponding to ω_{51}.

Figure 5.5: Spherical shell: First four unique (up to rotation) vibration modes using $p = 5$, $q = 5$ and $r = 2$. The colors corresponds to displacement in the radial direction.

Chapter 6

Exterior Helmholtz problems

In this chapter we shall consider the discretization of the fluid on an unbounded domain. As the finite element method is based on boundary value problems, we need to introduce an artificial boundary Γ_a (cf. Figure 6.1) where some sort of absorbing boundary conditions (ABC) must be implemented. One such method is the recently developed perfectly matched layer (PML) method after Bérenger (first introduced in [29] and [30]), where an additional layer is added outside of Γ_a to absorb outgoing plane waves at Γ_a. It is suggested that one can set $\Gamma_a = \Gamma$ (where Γ is the boundary of the elastic body) such that the FEM discretization of Ω_a is avoided. This would in turn reduce the computational time. In this chapter however, we shall implement the *infinite element* method for the domain Ω_a^+.

The exterior Helmholtz problem is given by

$$\begin{aligned} \Delta p + k^2 p &= 0 && \text{in} \quad \Omega^+, \\ \partial_n p &= g && \text{on} \quad \Gamma, \\ \frac{\partial p}{\partial R} - \mathrm{i}kp &= o(R^{-1}) && R \to \infty, \end{aligned}$$

where the last equation is the *Sommerfeld condition* which insures that no waves are reflected from infinity (for details see [3]).

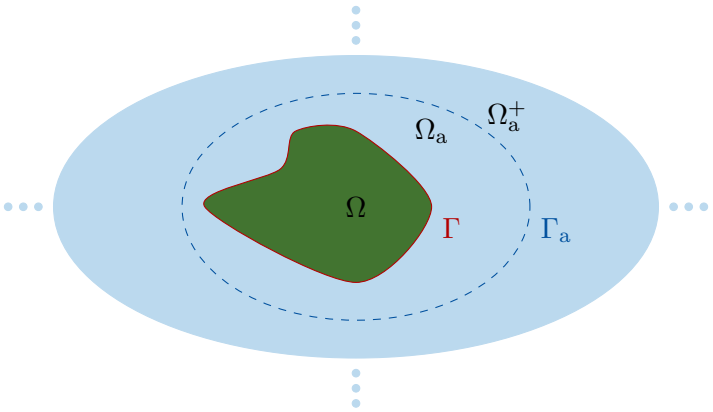


Figure 6.1: An artificial boundary Γ_a is introduced such that the exterior domain Ω^+ is decomposed by the two domains Ω_a (which is bounded by Γ and Γ_a) and Ω_a^+. Thus, $\Omega^+ = \Omega_a \cup \Omega_a^+$.

6.1 Weak formulation for the Helmholtz equation

As before, the weak form of the Helmholtz equation (which we have written in strong form) is found by multiplying with a test function and integration over the domain.

$$\int_{\Omega^+} q\Delta p + k^2 qp \,\mathrm{d}\Omega = 0$$

Using Equation (A.4) we get

$$-\int_{\Omega^+} \nabla q \cdot \nabla p \,\mathrm{d}\Omega + \int_{\partial\Omega^+} q\nabla p \cdot \boldsymbol{n} \,\mathrm{d}\Gamma + k^2 \int_{\Omega^+} qp \,\mathrm{d}\Omega = 0.$$

Thus,

$$\int_{\Omega^+} \nabla q \cdot \nabla p \,\mathrm{d}\Omega - k^2 \int_{\Omega^+} qp \,\mathrm{d}\Omega = \int_{\partial\Omega^+} qg \,\mathrm{d}\Gamma \tag{6.1}$$

The weak formulation thus becomes: Find $p \in H_w^{1+}(\Omega^+)$ such that

$$b(q,p) = L(q), \qquad \forall q \in H_{w^*}^1(\Omega^+)$$

where

$$b(q,p) = \int_{\Omega^+} \nabla q \cdot \nabla p \,\mathrm{d}\Omega - k^2 \int_{\Omega^+} qp \,\mathrm{d}\Omega$$

and

$$L(q) = \int_{\partial\Omega^+} qg \,\mathrm{d}\Gamma.$$

The reason why we no longer can use the space $H^1(\Omega^+)$ for the trial and the test space, is because Ω^+ is unbounded. Before stating the definitions of these spaces, one needs to understand the asymptotically behavior of the solution p at large r. In [31] Wilcox shows that the scalar field $p(\boldsymbol{r})$ satisfying the Helmholtz equation and the Sommerfield radiation conditions can be written on the form

$$p(\boldsymbol{r}) = \frac{\mathrm{e}^{\mathrm{i}kr}}{r} \sum_{n=0}^{\infty} \frac{p_n(\theta,\phi)}{r^n}$$

which implies that $|p| = \mathcal{O}(r^{-1})$ asymptotically for large r. Considering a function which represents this asymptotic property

$$f = \frac{\mathrm{e}^{\mathrm{i}kr}}{r}$$

we observe that the L^2 inner product does not exist. Indeed, if Γ is the unit sphere then

$$(f,f)_{L^2} = \int_{\Omega^+} \frac{\mathrm{e}^{\mathrm{i}kr}}{r} \frac{\mathrm{e}^{-\mathrm{i}kr}}{r} \,\mathrm{d}\Omega = 4\pi \int_1^\infty \frac{1}{r^2} r^2 \,\mathrm{d}r,$$

which is not finite. The solution to the problem is to introduce weighted norms by defining the inner product

$$(p,q)_w = \int_{\Omega^+} wp\bar{q} \,\mathrm{d}\Omega, \qquad \text{with} \quad w = \frac{1}{r^2}.$$

We may then define the norm

$$\|p\|_{1,w} = \sqrt{(p,p)_w + (\nabla p, \nabla p)_w}$$

such that the trial functions should satisfy $\|p\|_{1,w} < \infty$. The integrals

$$\int_{\Omega^+} p\bar{q}\,\mathrm{d}\Omega \quad \text{and} \quad \int_{\Omega^+} \nabla p \nabla \bar{q}\,\mathrm{d}\Omega$$

will be well defined if the test functions q are such that

$$(q,q)_{w^*} < \infty, \quad \text{and} \quad (\nabla q, \nabla q)_{w^*} < \infty$$

with the inner product

$$(p,q)_{w^*} = \int_{\Omega^+} w^* p\bar{q}\,\mathrm{d}\Omega, \qquad \text{with} \quad w^* = r^2.$$

and corresponding norm

$$\|p\|_{1,w^*} = \sqrt{(p,p)_{w^*} + (\nabla p, \nabla p)_{w^*}}$$

We thus define the *weighted Sobolev spaces*

$$H^1_w(\Omega^+) = \{p \,:\, \|p\|_{1,w} < \infty\}, \quad \text{and} \quad H^1_{w^*}(\Omega^+) = \{q \,:\, \|q\|_{1,w^*} < \infty\}$$

Hence, the functions in the trial space are of order r^{-1} (or lower), in contrast to the functions in the test space which are at most of order r^{-3}.

These definitions will not ensure any trial function satisfy the Sommerfeld condition. Leis solve this problem in [32] by modifying the trial space to be

$$H^{1+}_w(\Omega^+) = \{p \,:\, \|p\|^+_{1,w} < \infty\}$$

where

$$\|p\|^+_{1,w} = \sqrt{\|p\|^2_{1,w} + \int_{\Omega^+} \left|\frac{\partial p}{\partial r} - \mathrm{i}kp\right|^2 \mathrm{d}\Omega}.$$

For a more detailed discussion of the functional analysis involved in these spaces we refer to [3, pp. 41-43].

6.2 Infinite elements

In the following, we shall follow [10] in deriving the weak formulation for infinite elements using a prolate spheroidal coordinate system. We shall present four infinite element formulations; Petrov-Galerkin conjugated, Petrov-Galerkin unconjugated, Bubnov-Galerkin conjugated and Bubnov-Galerkin unconjugated (we shall refer to these methods as PGC, PGU, BGC and BGU). The Petrov-Galerkin formulations are based on the weighted Sobolev spaces after Leis [32]. It turns out that it is possible to create Bubnov-Galerkin formulations as well when the integration in the weak formulation is understood in the sense of the Cauchy principal value (consider [33] and [10] for details). The difference

between the conjugated formulations and the unconjugated formulations is simply conjugations of the test functions in the weak formulation. The formulations are heavily based on a prolate spheroidal coordinate system which is described in Appendix B.4 (this appendix should be studied to understand details and notation in this section).

We shall first develop the weak formulation for the unconjugated cases (PGU and BGU), and then present the conjugated cases (PGC and BGC).

The radial "shape functions" are given by

$$\phi_n(r) = \frac{\mathrm{e}^{\mathrm{i}kr}}{r^n}, \quad n = 1, \ldots, N. \tag{6.2}$$

The weak formulating takes the form[1]: For all $q \in V_2$, find $p_h^N \in V_1$ such that

$$b_{\mathrm{uc}}(q, p_h^N,) = \langle g, q \rangle_\Gamma,$$

where

$$b_{\mathrm{uc}}(q,p) = \lim_{\gamma\to\infty}\left(\int_{\Omega_\gamma}(\nabla q\nabla p - k^2 qp)\,\mathrm{d}\Omega - \int_{S_{\hat{r}}} q\partial_n p\,\mathrm{d}\Gamma\right), \tag{6.3}$$

$$\langle g, q\rangle_\Gamma = \int_\Gamma gq\,\mathrm{d}\Gamma.$$

Here, $S_{\hat{r}}$ is the surface where $r = \hat{r}$ and we can then recover the full domain by letting $\hat{r} \to \infty$. For the domain outside the artificial boundary ($r = r_{\mathrm{a}}$) we consider trial and test functions of the form

$$q = \frac{\mathrm{e}^{\mathrm{i}kr}}{r^n} f_n(\theta,\phi), \quad p = \frac{\mathrm{e}^{\mathrm{i}kr}}{r^m} f_m(\theta,\phi).$$

where the summation convention over the indices $n = 1, \ldots, N$ and $m = 1, \ldots, N$ is used for BGC and BGU and over the indices $n = 3, \ldots, N+2$ and $m = 1, \ldots, N$ for PGC and PGU.

Then, using the expression for the nabla operator found in Appendix B.4 we get

$$\begin{aligned}
\nabla q \cdot \nabla p =& \frac{1}{h_{\mathrm{r}}^2}\frac{\partial q}{\partial r}\frac{\partial p}{\partial r} + \frac{1}{h_\theta^2}\frac{\partial q}{\partial \theta}\frac{\partial p}{\partial \theta} + \frac{1}{h_\phi^2}\frac{\partial q}{\partial \phi}\frac{\partial p}{\partial \phi} \\
=& \left(\mathrm{i}k - \frac{n}{r}\right)\left(\mathrm{i}k - \frac{m}{r}\right)\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}h_{\mathrm{r}}^2} f_n f_m + \frac{\mathrm{e}^{2\mathrm{i}kr}}{h_\theta^2 r^{n+m}}\frac{\partial f_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{\mathrm{e}^{2\mathrm{i}kr}}{h_\phi^2 r^{n+m}}\frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi} \\
=& \frac{(r^2 - f^2)(-(kr)^2 - ikr(n+m) + nm)}{r^2(r^2 - f^2\cos^2\theta)}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}} f_n f_m \\
&+ \frac{1}{r^2 - f^2\cos^2\theta}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\frac{\partial f_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{1}{(r^2-f^2)\sin^2\theta}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}
\end{aligned}$$

which multiplied with the Jacobian J_1 yields

$$\begin{aligned}
\nabla q \cdot \nabla p J_1 =& \frac{(r^2 - f^2)(-(kr)^2 - ikr(n+m) + nm)}{r^2}\sin\theta\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}} f_n f_m \\
&+ \sin\theta\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\frac{\partial f_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{(r^2 - f^2\cos^2\theta)}{(r^2-f^2)\sin\theta}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}
\end{aligned}$$

[1] We refer to [3, pp. 89-90] for the precise definition of the spaces V_1 and V_2.

Also note that the term contributing to the mass matrix multiplied with the same Jacobian yields

$$k^2 pqJ_1 = \frac{k^2 \mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}(r^2 - f^2\cos^2\theta)\sin\theta f_n f_m.$$

Consider first the boundary integral at $S_{\hat{r}}$. In the limit $\hat{r} \to \infty$ we know that $\partial_n p \to \partial_r p$. Thus,

$$\begin{aligned}\int_{S_{\hat{r}}} q\partial_n p \,\mathrm{d}\Gamma &= \int_0^{2\pi}\int_0^{\pi} \left(\mathrm{i}k - \frac{m}{\hat{r}}\right)\frac{\mathrm{e}^{2\mathrm{i}kr}}{\hat{r}^{m+n}} f_n f_m \hat{r}^2 \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ &= \left(\mathrm{i}k - \frac{m}{\hat{r}}\right)\frac{1}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi\end{aligned}$$

As $m + n > 1$ all terms of order $\mathcal{O}(\hat{r}^{-(m+n-1)})$ vanish in the limit $\hat{r} \to \infty$, such that the integral reduces to

$$\int_{S_{\hat{r}}} q\partial_n p \,\mathrm{d}\Gamma = \frac{\mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi$$

Combining all of this into Equation (6.3) yields

$$\begin{aligned} b_{\mathrm{uc}}(q,p) = \lim_{\hat{r}\to\infty} \Bigg\{ &\int_{r_\mathrm{a}}^{\hat{r}} \left[-\frac{k^2}{r^{n+m-2}} - \frac{\mathrm{i}k(n+m)}{r^{n+m-1}} + \frac{nm + f^2k^2}{r^{n+m}} \right. \\ &\left. + \frac{\mathrm{i}kf^2(n+m)}{r^{n+m+1}} - \frac{nmf^2}{r^{n+m+2}} \right] \mathrm{e}^{2\mathrm{i}kr}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ &+ \int_{r_\mathrm{a}}^{\hat{r}} \frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\theta}\frac{\partial f_m}{\partial\theta}\sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ &+ \int_{r_\mathrm{a}}^{\hat{r}} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2 - f^2)r^{n+m-2}}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{1}{\sin\theta}\,\mathrm{d}\theta \,\mathrm{d}\phi \\ &- \int_{r_\mathrm{a}}^{\hat{r}} \frac{\mathrm{e}^{2\mathrm{i}kr} f^2}{(r^2 - f^2)r^{n+m}}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{\cos^2\theta}{\sin\theta}\,\mathrm{d}\theta \,\mathrm{d}\phi \\ &- \int_{r_\mathrm{a}}^{\hat{r}} \frac{k^2\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m-2}}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ &+ \int_{r_\mathrm{a}}^{\hat{r}} \frac{k^2 f^2 \mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\,\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} f_n f_m \cos^2\theta \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ &- \frac{\mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \Bigg\}. \end{aligned}$$

Defining the angular integrals

$$\begin{aligned} J_{mn}^{(1)} &= \int_0^{2\pi}\int_0^{\pi} f_n f_m \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi, & J_{mn}^{(2)} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\theta}\frac{\partial f_m}{\partial\theta}\sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \\ J_{mn}^{(3)} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{1}{\sin\theta}\,\mathrm{d}\theta \,\mathrm{d}\phi, & J_{mn}^{(4)} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial f_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{\cos^2\theta}{\sin\theta}\,\mathrm{d}\theta \,\mathrm{d}\phi \\ J_{mn}^{(5)} &= \int_0^{2\pi}\int_0^{\pi} f_n f_m \cos^2\theta \sin\theta \,\mathrm{d}\theta \,\mathrm{d}\phi \end{aligned} \tag{6.4}$$

we can write

$$b_{\mathrm{uc}}(q,p) = \lim_{\hat{r}\to\infty}\left\{ J^{(1)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\left[-\frac{k^2}{r^{n+m-2}} - \frac{\mathrm{i}k(n+m)}{r^{n+m-1}} + \frac{nm+f^2k^2}{r^{n+m}} \right.\right.$$
$$\left. + \frac{\mathrm{i}kf^2(n+m)}{r^{n+m+1}} - \frac{nmf^2}{r^{n+m+2}}\right]\mathrm{e}^{2\mathrm{i}kr}\,\mathrm{d}r + J^{(2)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\,\mathrm{d}r$$
$$+ J^{(3)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^{n+m-2}}\,\mathrm{d}r - J^{(4)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{\mathrm{e}^{2\mathrm{i}kr}f^2}{(r^2-f^2)r^{n+m}}\,\mathrm{d}r$$
$$\left. - J^{(1)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{k^2\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m-2}}\,\mathrm{d}r + J^{(5)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{k^2f^2\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m}}\,\mathrm{d}r - J^{(1)}_{mn}\frac{\mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}}{\hat{r}^{m+n-2}}\right\}.$$

In the case of Petrov-Galerkin formulations, all of these integrals exist as $n+m>3$. For the Bubnov Galerkin formulations, we must consider the special case $n=m=1$ (the integrals also exist for $n+m=3$). We collect the problematic terms in the following limit

$$L = \lim_{\hat{r}\to\infty}\left[J^{(1)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}} -\frac{k^2\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m-2}}\,\mathrm{d}r - J^{(1)}_{mn}\int_{r_\mathrm{a}}^{\hat{r}}\frac{k^2\mathrm{e}^{2\mathrm{i}kr}}{r^{n+m-2}}\,\mathrm{d}r - J^{(1)}_{mn}\frac{\mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}}{\hat{r}^{m+n-2}}\right].$$

Inserting $n=m=1$ and performing integration yields

$$\begin{aligned} L &= J^{(1)}_{11}\lim_{\hat{r}\to\infty}\left[-2\int_{r_\mathrm{a}}^{\hat{r}}k^2\mathrm{e}^{2\mathrm{i}kr}\,\mathrm{d}r - \mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}\right] \\ &= J^{(1)}_{11}\lim_{\hat{r}\to\infty}\left[\mathrm{i}k\left(\mathrm{e}^{2\mathrm{i}k\hat{r}} - \mathrm{e}^{2\mathrm{i}kr_\mathrm{a}}\right) - \mathrm{i}k\mathrm{e}^{2\mathrm{i}k\hat{r}}\right] \\ &\quad . \end{aligned}$$

which exist due to cancellation of oscillatory terms. Thus, the limit is

$$L = -J^{(1)}_{11}\mathrm{i}k\mathrm{e}^{2\mathrm{i}kr_\mathrm{a}}.$$

Integrals of the form

$$\int_{r_\mathrm{a}}^{\infty}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^n}\,\mathrm{d}r = \frac{1}{r_\mathrm{a}^{n-1}}\int_1^{\infty}\frac{\mathrm{e}^{2\mathrm{i}kr_\mathrm{a}z}}{z^n}\,\mathrm{d}z, \quad n\geq 1$$

exist and may be computed. Indeed, the *exponential integral* function defined by

$$E_n(x) = \int_1^{\infty}\frac{\mathrm{e}^{-xz}}{z^n}\,\mathrm{d}z$$

is implemented in MATLAB for $n=1$, and using the recursive relation

$$E_n(x) = \frac{1}{n-1}\left(\mathrm{e}^{-x} - xE_{n-1}(x)\right), \quad n=2,3,4,\ldots$$

found in [13, p. 229], we can compute the integrals by

$$\int_{r_\mathrm{a}}^{\infty}\frac{\mathrm{e}^{2\mathrm{i}kr}}{r^n}\,\mathrm{d}r = \frac{1}{r_\mathrm{a}^{n-1}}E_n(-2\mathrm{i}kr_\mathrm{a}).$$

Writing out the recursive relation, one may also use the explicit formula

$$E_n(x) = \frac{(-x)^{n-1}}{(n-1)!}E_1(x) + \mathrm{e}^{-x}\sum_{m=0}^{n-2}\frac{(-x)^m}{\mathrm{nPk(n-1,m+1)}} \tag{6.5}$$

where

$$\mathrm{nPk}(n,k) = \frac{n!}{(n-k)!}.$$

The other type of integrals are of the form

$$\int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^n}\,\mathrm{d}r, \quad n \geq 0. \tag{6.6}$$

One can show by induction that (for $n \geq 1$)

$$\frac{1}{(r^2-f^2)r^n} = \begin{cases} \frac{1}{f^{n+1}}\frac{r}{r^2-f^2} - \sum_{j=0}^{\frac{n-1}{2}} \frac{1}{f^{2(j+1)}r^{n-2j}} & \text{for odd } n \\ \frac{1}{f^n}\frac{1}{r^2-f^2} - \sum_{j=0}^{\frac{n-2}{2}} \frac{1}{f^{2(j+1)}r^{n-2j}} & \text{for even } n \end{cases}$$

where

$$\frac{r}{r^2-f^2} = \frac{1}{2}\left(\frac{1}{r-f}+\frac{1}{r+f}\right) \quad \text{and} \quad \frac{1}{r^2-f^2} = \frac{1}{2f}\left(\frac{1}{r-f}-\frac{1}{r+f}\right).$$

As

$$\int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{r\pm f}\,\mathrm{d}r = \mathrm{e}^{\mp 2\mathrm{i}kf}\int_{1}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}k(r_\mathrm{a}\pm f)z}}{z}\,\mathrm{d}z = \mathrm{e}^{\mp 2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}\pm f))$$

we can write the integrals in Equation (6.6) as a series of exponential integrals. Indeed, for n odd we have the formula

$$\begin{aligned}\int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^n}\,\mathrm{d}r =& \frac{1}{2f^{n+1}}\left[\mathrm{e}^{2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}-f)) + \mathrm{e}^{-2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}+f))\right] \\ &- \sum_{j=0}^{\frac{n-1}{2}} \frac{1}{f^{2(j+1)}}\frac{1}{r_\mathrm{a}^{n-2j-1}}E_{n-2j}(-2\mathrm{i}kr_\mathrm{a})\end{aligned}$$

and for n even we get

$$\begin{aligned}\int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^n}\,\mathrm{d}r =& \frac{1}{2f^{n+1}}\left[\mathrm{e}^{2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}-f)) - \mathrm{e}^{-2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}+f))\right] \\ &- \sum_{j=0}^{\frac{n-2}{2}} \frac{1}{f^{2(j+1)}}\frac{1}{r_\mathrm{a}^{n-2j-1}}E_{n-2j}(-2\mathrm{i}kr_\mathrm{a}).\end{aligned}$$

where we have included the case $n = 0$ (where we interpret the sum to be zero). More compactly, we may write (for all $n \geq 0$)

$$\begin{aligned}\int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^n}\,\mathrm{d}r =& \frac{1}{2f^{n+1}}\left[\mathrm{e}^{2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}-f)) - (-1)^n\mathrm{e}^{-2\mathrm{i}kf}E_1(-2\mathrm{i}k(r_\mathrm{a}+f))\right] \\ &- \sum_{j=0}^{\lfloor\frac{n-1}{2}\rfloor} \frac{1}{f^{2(j+1)}}\frac{1}{r_\mathrm{a}^{n-2j-1}}E_{n-2j}(-2\mathrm{i}kr_\mathrm{a}).\end{aligned}$$

Using Equation (6.5), it is possible to write the last sum as

$$\begin{aligned}\sum_{j=0}^{\lfloor\frac{n-1}{2}\rfloor} \frac{1}{f^{2(j+1)}}\frac{1}{r_\mathrm{a}^{n-2j-1}}E_{n-2j}(-2\mathrm{i}kr_\mathrm{a}) =& E_1(-2\mathrm{i}kr_\mathrm{a})\sum_{j=0}^{\lfloor\frac{n-1}{2}\rfloor} \frac{1}{f^{2(j+1)}}\frac{(2\mathrm{i}k)^{n-2j-1}}{(n-2j-1)!} \\ &+ \mathrm{e}^{2\mathrm{i}kr_\mathrm{a}}\sum_{j=0}^{\lfloor\frac{n-1}{2}\rfloor}\sum_{m=0}^{n-2j-2} \frac{1}{f^{2(j+1)}}\frac{(2\mathrm{i}k)^m r_\mathrm{a}^{m+2j-n+1}}{\mathrm{nPk}(n-2j-1,m+1)}.\end{aligned}$$

For large n, this formula suffers from high precision loss when implemented with a given precision. This is because the two terms in the last expression partly cancel each other out. One thus needs to increase the floating point precision correspondingly for an increase in n. In MATLAB, this could be done by invoking MuPAD.

Using the following notation for the radial integrals

$$I_n^{(1)} = \int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{r^n}\,\mathrm{d}r \qquad I_n^{(2)} = \int_{r_\mathrm{a}}^{\infty} \frac{\mathrm{e}^{2\mathrm{i}kr}}{(r^2-f^2)r^{n-1}}\,\mathrm{d}r, \qquad n \geq 1$$

we may now write the bilinear form as (for $n+m>2$)

$$\begin{aligned} b_\mathrm{uc}(q,p) =& J_{mn}^{(1)}\left[-2k^2 I_{n+m-2}^{(1)} - \mathrm{i}k(n+m)I_{n+m-1}^{(1)} + (nm+f^2k^2)I_{n+m}^{(1)}\right.\\ &\left.+\mathrm{i}kf^2(n+m)I_{n+m+1}^{(1)} - nmf^2 I_{n+m+2}^{(1)}\right] + J_{mn}^{(2)} I_{n+m}^{(1)}\\ &+ J_{mn}^{(3)} I_{n+m-1}^{(2)} - J_{mn}^{(4)} f^2 I_{n+m+1}^{(2)} + J_{mn}^{(5)} k^2 f^2 I_{n+m}^{(1)}. \end{aligned}$$

and for $n=m=1$ we get

$$\begin{aligned} b_\mathrm{uc}(q,p) =& J_{11}^{(1)}\left[-2\mathrm{i}kI_1^{(1)} + (1+f^2k^2)I_2^{(1)} + 2\mathrm{i}kf^2 I_3^{(1)} - f^2 I_4^{(1)} - \mathrm{i}k\mathrm{e}^{2\mathrm{i}kr_\mathrm{a}}\right]\\ &+ J_{11}^{(2)} I_2^{(1)} + J_{11}^{(3)} I_1^{(2)} - J_{11}^{(4)} f^2 I_3^{(2)} + J_{11}^{(5)} k^2 f^2 I_2^{(1)}. \end{aligned}$$

Let

$$f_n = \sum_{A\in\boldsymbol{\eta}_\mathrm{a}} \mathrm{e}^{-\mathrm{i}kr_\mathrm{a}} r_\mathrm{a}^n R_A c_{nA}, \quad f_m = \sum_{B\in\boldsymbol{\eta}_\mathrm{a}} \mathrm{e}^{-\mathrm{i}kr_\mathrm{a}} r_\mathrm{a}^m R_B d_{mB}$$

where $\boldsymbol{\eta}_\mathrm{a}$ is the set containing the indices of all the basis functions that are non-zero on Γ_a. Note that the scaling factors $\mathrm{e}^{-\mathrm{i}kr_\mathrm{a}} r_\mathrm{a}^n$ and $\mathrm{e}^{-\mathrm{i}kr_\mathrm{a}} r_\mathrm{a}^m$ must be used to get continuous basis functions (as the NURBS basis functions satisfy the unity property inside of Γ_a, so must the semi analytic basis functions outside of Γ_a). Using the bilinearity of b_uc we obtain

$$\sum_{A\in\boldsymbol{\eta}_\mathrm{a}}\sum_{n=1}^{N} c_{nA}\left(\sum_{B\in\boldsymbol{\eta}_\mathrm{a}}\sum_{m=1}^{N} \mathrm{e}^{-2\mathrm{i}kr_\mathrm{a}} r_\mathrm{a}^{m+n} d_{mB} b_\mathrm{uc}(R_A,R_B)\right) = 0.$$

As the surface integrals in Equation (6.4) will now be independent of the indices m and n we now get (for $n+m>1$)

$$\begin{aligned} b_\mathrm{uc}(R_A,R_B) =& J_{AB}^{(1)}\left[-2k^2 I_{n+m-2}^{(1)} - \mathrm{i}k(n+m)I_{n+m-1}^{(1)} + (nm+f^2k^2)I_{n+m}^{(1)}\right.\\ &\left.+\mathrm{i}kf^2(n+m)I_{n+m+1}^{(1)} - nmf^2 I_{n+m+2}^{(1)}\right] + J_{AB}^{(2)} I_{n+m}^{(1)}\\ &+ J_{AB}^{(3)} I_{n+m-1}^{(2)} - J_{AB}^{(4)} f^2 I_{n+m+1}^{(2)} + J_{AB}^{(5)} k^2 f^2 I_{n+m}^{(1)}. \end{aligned}$$

and in the case $n=m=1$

$$\begin{aligned} b_\mathrm{uc}(R_A,R_B) =& J_{AB}^{(1)}\left[-2\mathrm{i}kI_1^{(1)} + (1+f^2k^2)I_2^{(1)} + 2\mathrm{i}kf^2 I_3^{(1)} - f^2 I_4^{(1)} - \mathrm{i}k\mathrm{e}^{2\mathrm{i}kr_\mathrm{a}}\right]\\ &+ J_{AB}^{(2)} I_2^{(1)} + J_{AB}^{(3)} I_1^{(2)} - J_{AB}^{(4)} f^2 I_3^{(2)} + J_{AB}^{(5)} k^2 f^2 I_2^{(1)}. \end{aligned}$$

where we have redefined the surface integral notations to be

$$J_{AB}^{(1)} = \int_0^{2\pi}\int_0^{\pi} R_A R_B \sin\theta \,\mathrm{d}\theta\,\mathrm{d}\phi, \qquad J_{AB}^{(2)} = \int_0^{2\pi}\int_0^{\pi} \frac{\partial R_A}{\partial\theta}\frac{\partial R_B}{\partial\theta}\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi$$
$$J_{AB}^{(3)} = \int_0^{2\pi}\int_0^{\pi} \frac{\partial R_A}{\partial\phi}\frac{\partial R_B}{\partial\phi}\frac{1}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi, \qquad J_{AB}^{(4)} = \int_0^{2\pi}\int_0^{\pi} \frac{\partial R_A}{\partial\phi}\frac{\partial R_B}{\partial\phi}\frac{\cos^2\theta}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi \quad (6.7)$$
$$J_{AB}^{(5)} = \int_0^{2\pi}\int_0^{\pi} R_A R_B \cos^2\theta\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi$$

The infinite elements are thus only contributing whenever we have an element adjacent to the boundary Γ_a. It is important to note that we now have increased the number of degrees of freedom at the surface Γ_a from $|\boldsymbol{\eta}_\mathrm{a}|$ to $N|\boldsymbol{\eta}_\mathrm{a}|$. Hence, the method of infinite elements adds a total of $(N-1)|\boldsymbol{\eta}_\mathrm{a}|$ degrees of freedom. Due to redundancy, we should not evaluate the full 3D NURBS function set at $\zeta = 1$ as the 3D NURBS basis functions reduce to a 2D NURBS surface in 3D. Rather, we loop over the surface elements which is then done separate from the main matrix assembly. Thus, the procedure of adding contribution from infinite elements follows the procedure of applying Neumann conditions except that we do not update the load vector $\boldsymbol{F}$, rather, we now update the global matrix. As the artificial boundary Γ_a at $\zeta = 1$ (which is at r_a in the prolate spheroidal coordinates) may be parametrized by both (ξ,η) and (θ,ϕ) we have

$$\mathrm{d}\theta\mathrm{d}\phi = \begin{vmatrix} \frac{\partial\theta}{\partial\xi} & \frac{\partial\phi}{\partial\xi} \\ \frac{\partial\theta}{\partial\eta} & \frac{\partial\phi}{\partial\eta} \end{vmatrix} \mathrm{d}\xi\mathrm{d}\eta \quad (6.8)$$

where

$$\frac{\partial\theta}{\partial\xi} = \frac{\partial\theta}{\partial x}\frac{\partial x}{\partial\xi} + \frac{\partial\theta}{\partial y}\frac{\partial y}{\partial\xi} + \frac{\partial\theta}{\partial z}\frac{\partial z}{\partial\xi}, \qquad \frac{\partial\theta}{\partial\eta} = \frac{\partial\theta}{\partial x}\frac{\partial x}{\partial\eta} + \frac{\partial\theta}{\partial y}\frac{\partial y}{\partial\eta} + \frac{\partial\theta}{\partial z}\frac{\partial z}{\partial\eta}$$
$$\frac{\partial\phi}{\partial\xi} = \frac{\partial\phi}{\partial x}\frac{\partial x}{\partial\xi} + \frac{\partial\phi}{\partial y}\frac{\partial y}{\partial\xi} + \frac{\partial\phi}{\partial z}\frac{\partial z}{\partial\xi}, \qquad \frac{\partial\phi}{\partial\eta} = \frac{\partial\phi}{\partial x}\frac{\partial x}{\partial\eta} + \frac{\partial\phi}{\partial y}\frac{\partial y}{\partial\eta} + \frac{\partial\phi}{\partial z}\frac{\partial z}{\partial\eta}$$

This may be written more densely by

$$\begin{bmatrix} \frac{\partial\theta}{\partial\xi} & \frac{\partial\theta}{\partial\eta} \\ \frac{\partial\phi}{\partial\xi} & \frac{\partial\phi}{\partial\eta} \end{bmatrix} = \begin{bmatrix} \frac{\partial\theta}{\partial x} & \frac{\partial\theta}{\partial y} & \frac{\partial\theta}{\partial z} \\ \frac{\partial\phi}{\partial x} & \frac{\partial\phi}{\partial y} & \frac{\partial\phi}{\partial z} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial\xi} & \frac{\partial x}{\partial\eta} \\ \frac{\partial y}{\partial\xi} & \frac{\partial y}{\partial\eta} \\ \frac{\partial z}{\partial\xi} & \frac{\partial z}{\partial\eta} \end{bmatrix} \quad (6.9)$$

where the partial derivatives with respect to the coordinate transformation is found in Equation (B.34).

The derivatives of the basis functions may be computed by

$$\begin{bmatrix} \frac{\partial R_A}{\partial\theta} \\ \frac{\partial R_A}{\partial\phi} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial\theta} & \frac{\partial y}{\partial\theta} & \frac{\partial z}{\partial\theta} \\ \frac{\partial x}{\partial\phi} & \frac{\partial y}{\partial\phi} & \frac{\partial z}{\partial\phi} \end{bmatrix} \begin{bmatrix} \frac{\partial R_A}{\partial x} \\ \frac{\partial R_A}{\partial y} \\ \frac{\partial R_A}{\partial z} \end{bmatrix} \quad (6.10)$$

where the inverse partial derivatives with respect to the coordinate transformation is found in Equation (B.33). However, as we shall parametrize the prolate spheroid such that $\xi = \xi(\phi)$ and $\eta = \eta(\theta)$, we can simplify the formulas. As $R_A = R_A(\xi,\eta)$ at the surface Γ_a (that is, the basis functions are independent of ζ) we get

$$\frac{\partial R_A}{\partial\theta} = \frac{\partial R_A}{\partial\xi}\frac{\mathrm{d}\xi}{\mathrm{d}\theta} + \frac{\partial R_A}{\partial\eta}\frac{\mathrm{d}\eta}{\mathrm{d}\theta}$$

where

$$\frac{\mathrm{d}\xi}{\mathrm{d}\theta} = 0 \quad \text{and} \quad \frac{\mathrm{d}\eta}{\mathrm{d}\theta} = \left(\frac{\mathrm{d}\theta}{\mathrm{d}\eta}\right)^{-1}.$$

With a corresponding formula for the derivative with respect to ϕ we therefore have

$$\frac{\partial R_A}{\partial \theta} = \frac{\partial R_A}{\partial \eta}\left(\frac{\mathrm{d}\theta}{\mathrm{d}\eta}\right)^{-1} \quad \text{and} \quad \frac{\partial R_A}{\partial \phi} = \frac{\partial R_A}{\partial \xi}\left(\frac{\mathrm{d}\phi}{\mathrm{d}\xi}\right)^{-1}. \tag{6.11}$$

With such a parametrization the four equations in Equation (6.9) reduces to

$$\begin{bmatrix} 0 & \frac{\mathrm{d}\theta}{\mathrm{d}\eta} \\ \frac{\mathrm{d}\phi}{\mathrm{d}\xi} & 0 \end{bmatrix} = \begin{bmatrix} \frac{\partial\theta}{\partial x} & \frac{\partial\theta}{\partial y} & \frac{\partial\theta}{\partial z} \\ \frac{\partial\phi}{\partial x} & \frac{\partial\phi}{\partial y} & \frac{\partial\phi}{\partial z} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial\xi} & \frac{\partial x}{\partial\eta} \\ \frac{\partial y}{\partial\xi} & \frac{\partial y}{\partial\eta} \\ \frac{\partial y}{\partial\xi} & \frac{\partial y}{\partial\eta} \end{bmatrix} \tag{6.12}$$

It is also possible to use the complex conjugate of the test functions. Instead of a bilinear form, we then get a sesquilinear form. The weak formulating then takes the form[2]: For all $q \in V_2$, find $p_h^N \in V_1$ such that

$$b_{\mathrm{c}}(q, p_h^N) = \langle g, q\rangle_\Gamma,$$

where

$$b_{\mathrm{c}}(q,p) = \lim_{\gamma\to\infty}\left(\int_{\Omega_\gamma} (\nabla\bar{q}\nabla p - k^2\bar{q}p)\,\mathrm{d}\Omega - \int_{S_{\hat{r}}} \bar{q}\partial_n p\,\mathrm{d}\Gamma\right), \tag{6.13}$$

$$\langle g, \bar{q}\rangle_\Gamma = \int_\Gamma g\bar{q}\,\mathrm{d}\Gamma.$$

We now get

$$\begin{aligned}
\nabla\bar{q}\cdot\nabla p =& \frac{1}{h_{\mathrm{r}}^2}\frac{\partial\bar{q}}{\partial r}\frac{\partial p}{\partial r} + \frac{1}{h_{\theta}^2}\frac{\partial\bar{q}}{\partial\theta}\frac{\partial p}{\partial\theta} + \frac{1}{h_{\phi}^2}\frac{\partial\bar{q}}{\partial\phi}\frac{\partial p}{\partial\phi} \\
=& \left(-\mathrm{i}k - \frac{n}{r}\right)\left(\mathrm{i}k - \frac{m}{r}\right)\frac{1}{r^{n+m}h_{\mathrm{r}}^2}\bar{f}_n f_m + \frac{1}{h_{\theta}^2 r^{n+m}}\frac{\partial\bar{f}_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{1}{h_{\phi}^2 r^{n+m}}\frac{\partial\bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi} \\
=& \frac{(r^2-f^2)((kr)^2 + ikr(m-n) + nm)}{r^2(r^2 - f^2\cos^2\theta)}\frac{1}{r^{n+m}}\bar{f}_n f_m \\
&+ \frac{1}{r^2 - f^2\cos^2\theta}\frac{1}{r^{n+m}}\frac{\partial\bar{f}_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{1}{(r^2-f^2)\sin^2\theta}\frac{1}{r^{n+m}}\frac{\partial\bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}
\end{aligned}$$

which multiplied with the Jacobian J_1 yields

$$\begin{aligned}
\nabla\bar{q}\cdot\nabla p J_1 =& \frac{(r^2-f^2)((kr)^2 + ikr(m-n) + nm)}{r^2}\sin\theta\frac{1}{r^{n+m}}\bar{f}_n f_m \\
&+ \sin\theta\frac{1}{r^{n+m}}\frac{\partial\bar{f}_n}{\partial\theta}\frac{\partial f_m}{\partial\theta} + \frac{(r^2 - f^2\cos^2\theta)}{(r^2-f^2)\sin\theta}\frac{1}{r^{n+m}}\frac{\partial\bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}
\end{aligned}$$

[2] Again, we refer to [3, pp. 89-90] for the precise definition of the spaces V_1 and V_2.

Also note that the term contributing to the mass matrix multiplied with the same Jacobian yields

$$k^2 p\bar{q} J_1 = \frac{k^2}{r^{n+m}}(r^2 - f^2\cos^2\theta)\sin\theta \bar{f}_n f_m.$$

Consider first the boundary integral at $S_{\hat{r}}$. In the limit $\hat{r} \to \infty$ we know that $\partial_n p \to \partial_r p$ (as prolate spheroidal coordinates reduces to spherical coordinates in this limit). Thus,

$$\begin{aligned}\int_{S_{\hat{r}}} \bar{q}\partial_n u \,\mathrm{d}\Gamma &= \int_0^{2\pi}\int_0^{\pi} \left(\mathrm{i}k - \frac{m}{\hat{r}}\right)\frac{1}{\hat{r}^{m+n}}\bar{f}_n f_m \hat{r}^2 \sin\theta \,\mathrm{d}\theta\,\mathrm{d}\phi \\ &= \left(\mathrm{i}k - \frac{m}{\hat{r}}\right)\frac{1}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi\end{aligned}$$

As $m + n > 1$ all terms of order $\mathcal{O}(\hat{r}^{-(m+n-1)})$ vanish in the limit $\hat{r} \to \infty$, such that the integral reduces to

$$\int_{S_{\hat{r}}} q\partial_n p \,\mathrm{d}\Gamma = \frac{\mathrm{i}k}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi$$

Combining all of this into Equation (6.13) yields

$$\begin{aligned} b_{\mathrm{c}}(q,p) = \lim_{\hat{r}\to\infty} \Bigg\{ &\int_{r_{\mathrm{a}}}^{\hat{r}} \left[\frac{k^2}{r^{n+m-2}} + \frac{\mathrm{i}k(m-n)}{r^{n+m-1}} + \frac{nm - f^2k^2}{r^{n+m}} \right.\\ &\left. - \frac{\mathrm{i}kf^2(m-n)}{r^{n+m+1}} - \frac{nmf^2}{r^{n+m+2}}\right]\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &+ \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{1}{r^{n+m}}\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\theta}\frac{\partial f_m}{\partial\theta}\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &+ \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{1}{(r^2-f^2)r^{n+m-2}}\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{1}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &- \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{f^2}{(r^2-f^2)r^{n+m}}\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{\cos^2\theta}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &- \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{k^2}{r^{n+m-2}}\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &+ \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{k^2f^2}{r^{n+m}}\mathrm{d}r \int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \cos^2\theta\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &- \frac{\mathrm{i}k}{\hat{r}^{m+n-2}}\int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \Bigg\}. \end{aligned}$$

Defining the angular integrals

$$\begin{aligned} J^{(1)}_{mn} &= \int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi, & J^{(2)}_{mn} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\theta}\frac{\partial f_m}{\partial\theta}\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \\ J^{(3)}_{mn} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{1}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi, & J^{(4)}_{mn} &= \int_0^{2\pi}\int_0^{\pi} \frac{\partial \bar{f}_n}{\partial\phi}\frac{\partial f_m}{\partial\phi}\frac{\cos^2\theta}{\sin\theta}\,\mathrm{d}\theta\,\mathrm{d}\phi \\ J^{(5)}_{mn} &= \int_0^{2\pi}\int_0^{\pi} \bar{f}_n f_m \cos^2\theta\sin\theta\,\mathrm{d}\theta\,\mathrm{d}\phi \end{aligned} \tag{6.14}$$

and noticing cancellation of the two terms $\frac{k^2}{r^{n+m-2}}$ we can write

$$\begin{aligned} b_{\mathrm{c}}(q,p) = \lim_{\hat{r}\to\infty} \Bigg\{ & J^{(1)}_{mn} \int_{r_{\mathrm{a}}}^{\hat{r}} \left[\frac{\mathrm{i}k(m-n)}{r^{n+m-1}} + \frac{nm - f^2k^2}{r^{n+m}} \right. \\ & \left. - \frac{\mathrm{i}kf^2(m-n)}{r^{n+m+1}} - \frac{nmf^2}{r^{n+m+2}} \right] \mathrm{d}r + J^{(2)}_{mn} \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{1}{r^{n+m}} \,\mathrm{d}r \\ & + J^{(3)}_{mn} \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{1}{(r^2-f^2)r^{n+m-2}} \,\mathrm{d}r - J^{(4)}_{mn} \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{f^2}{(r^2-f^2)r^{n+m}} \,\mathrm{d}r \\ & + J^{(5)}_{mn} \int_{r_{\mathrm{a}}}^{\hat{r}} \frac{k^2f^2}{r^{n+m}} \,\mathrm{d}r - J^{(1)}_{mn} \frac{\mathrm{i}k}{\hat{r}^{m+n-2}} \Bigg\}. \end{aligned}$$

All of these integrals exists, also for the case $m = n = 1$ as the term $\mathcal{O}(r^{-(n+m-1)})$ vanish due to its numerator.

The radial integrals may be computed by

$$\int_{r_{\mathrm{a}}}^{\infty} \frac{1}{r^n} \,\mathrm{d}r = \frac{1}{r_{\mathrm{a}}^{n-1}} \frac{1}{n-1}, \quad n > 1$$

and by the use of Laurent series (as $f < r_{\mathrm{a}}$) we get

$$\int_{r_{\mathrm{a}}}^{\infty} \frac{1}{(r^2-f^2)r^n} \,\mathrm{d}r = \frac{1}{r_{\mathrm{a}}^{2+n-1}} \sum_{j=0}^{\infty} \left(\frac{f}{r_{\mathrm{a}}}\right)^{2j} \frac{1}{2j+n+1}.$$

Using the following notation for the radial integrals

$$\begin{aligned} I^{(1)}_n &= \int_{r_{\mathrm{a}}}^{\infty} \frac{1}{r^n} \,\mathrm{d}r, \qquad n \geq 2 \\ I^{(2)}_n &= \int_{r_{\mathrm{a}}}^{\infty} \frac{1}{(r^2-f^2)r^{n-1}} \,\mathrm{d}r, \qquad n \geq 1 \end{aligned}$$

we may now write the sesquilinear form as (for $n + m > 2$)

$$\begin{aligned} b_{\mathrm{c}}(q,p) = & J^{(1)}_{mn} \left[\mathrm{i}k(m-n) I^{(1)}_{n+m-1} + (nm - f^2k^2) I^{(1)}_{n+m} \right. \\ & \left. - \mathrm{i}kf^2(m-n) I^{(1)}_{n+m+1} - nmf^2 I^{(1)}_{n+m+2} \right] + J^{(2)}_{mn} I^{(1)}_{n+m} \\ & + J^{(3)}_{mn} I^{(2)}_{n+m-1} - J^{(4)}_{mn} f^2 I^{(2)}_{n+m+1} + J^{(5)}_{mn} k^2 f^2 I^{(1)}_{n+m}. \end{aligned}$$

and for $n = m = 1$ we get

$$\begin{aligned} b_{\mathrm{c}}(q,p) = & J^{(1)}_{11} \left[(1 - f^2k^2) I^{(1)}_2 - f^2 I^{(1)}_4 - \mathrm{i}k \right] \\ & + J^{(2)}_{11} I^{(1)}_2 + J^{(3)}_{11} I^{(2)}_1 - J^{(4)}_{11} f^2 I^{(2)}_3 + J^{(5)}_{11} k^2 f^2 I^{(1)}_2. \end{aligned}$$

Let

$$f_n = \sum_{A \in \boldsymbol{\eta}_{\mathrm{a}}} \mathrm{e}^{-\mathrm{i}kr_{\mathrm{a}}} r_{\mathrm{a}}^n R_A \bar{c}_{nA}, \quad f_m = \sum_{B \in \boldsymbol{\eta}_{\mathrm{a}}} \mathrm{e}^{-\mathrm{i}kr_{\mathrm{a}}} r_{\mathrm{a}}^m R_B d_{mB}$$

where (as before) $\boldsymbol{\eta}_{\mathrm{a}}$ is the set containing the indices of all the basis functions that are non-zero on Γ_{a}. Using the sequilinearity of b_{c} we obtain

$$\sum_{A \in \boldsymbol{\eta}_{\mathrm{a}}} \sum_{n=1}^{N} \bar{c}_{nA} \left(\sum_{B \in \boldsymbol{\eta}_{\mathrm{a}}} \sum_{m=1}^{N} r_{\mathrm{a}}^{n+m} d_{mB} b_{\mathrm{c}}(R_A, R_B) \right) = 0.$$

As the surface integrals in Equation (6.14) will now be independent of the indices m and n we now get (for $n+m>1$)

$$\begin{aligned} b_{\mathrm{c}}(R_A,R_B) =& J_{AB}^{(1)}\Big[\mathrm{i}k(m-n)I_{n+m-1}^{(1)} + (nm-f^2k^2)I_{n+m}^{(1)} \\ & -\mathrm{i}kf^2(m-n)I_{n+m+1}^{(1)} - nmf^2I_{n+m+2}^{(1)}\Big] + J_{AB}^{(2)}I_{n+m}^{(1)} \\ & + J_{AB}^{(3)}I_{n+m-1}^{(2)} - J_{AB}^{(4)}f^2I_{n+m+1}^{(2)} + J_{AB}^{(5)}k^2f^2I_{n+m}^{(1)}. \end{aligned}$$

and for $n=m=1$ we get

$$\begin{aligned} b_{\mathrm{c}}(R_A,R_B) =& J_{AB}^{(1)}\Big[(1-f^2k^2)I_2^{(1)} - f^2I_4^{(1)} - \mathrm{i}k\Big] \\ & + J_{AB}^{(2)}I_2^{(1)} + J_{AB}^{(3)}I_1^{(2)} - J_{AB}^{(4)}f^2I_3^{(2)} + J_{AB}^{(5)}k^2f^2I_2^{(1)}. \end{aligned}$$

where the redefined surface integrals are the same as in Equation (6.7).

6.2.1 The multipole expansion

In [10] Burnett presents a multipole expansion in which the radial basis function in the infinite elements have poles at different layers with radii r_n, $n=1,\ldots,N$. That is, instead of using the radial shape functions in Equation (6.2) we may use the following definition

$$\phi_n(r) = \mathrm{e}^{\mathrm{i}k(r-r_n)}\sum_{n'=1}^{N}\frac{H_{nn'}}{r^{n'}}, \quad n=1,\ldots,N.$$

where the coefficients $H_{nn'}$ are determined in order to satisfy

$$\phi_n(r_{n'}) = \delta_{nn'}.$$

Hence, the coefficients are determined from the relation

$$\boldsymbol{H}\boldsymbol{S} = \boldsymbol{I} \tag{6.15}$$

where

$$\boldsymbol{H} = \begin{bmatrix} H_{11} & H_{12} & \ldots & H_{1N} \\ H_{21} & H_{22} & \ldots & H_{2N} \\ \vdots & \vdots & \ldots & \vdots \\ H_{N1} & H_{N2} & \ldots & H_{NN} \end{bmatrix} \quad \text{and} \quad \boldsymbol{S} = \begin{bmatrix} r_1^{-1} & r_2^{-1} & \ldots & r_N^{-1} \\ r_1^{-2} & r_2^{-2} & \ldots & r_N^{-2} \\ \vdots & \vdots & \ldots & \vdots \\ r_1^{-N} & r_2^{-N} & \ldots & r_N^{-N} \end{bmatrix}.$$

and $\boldsymbol{I}$ is the $N\times N$ identity matrix. The coefficients $H_{nn'}$ are thus given by $\boldsymbol{H}=\boldsymbol{S}^{-1}$.

These basis functions differs from the classical FEM shape functions in that they have global support (outside the artificial boundary Γ_{a}), but only the first basis function, ϕ_1 are non zero at Γ_{a}. This property improves the sparsity of the global matrix, such that this method improves performance. The accuracy however, seems to be unchanged. This is to be expected as the basis functions spans the same space.

For the Petrov-Galerkin formulations, we must create a second set of basis functions (for the test space), namely

$$\psi_n(r) = \mathrm{e}^{\mathrm{i}k(r-r_n)}\sum_{n'=3}^{N+2}\frac{H'_{nn'}}{r^{n'}}, \quad n=3,\ldots,N+2.$$

The coefficients $H'_{nn'}$ are found in the same way as before but now with the matrix

$$\boldsymbol{S}' = \begin{bmatrix} r_1^{-3} & r_2^{-3} & \dots & r_N^{-3} \\ r_1^{-4} & r_2^{-4} & \dots & r_N^{-4} \\ \vdots & \vdots & \dots & \vdots \\ r_1^{-N-2} & r_2^{-N-2} & \dots & r_N^{-N-2} \end{bmatrix}.$$

instead of $\boldsymbol{S}$. It should be noted that the condition number of the matrices $\boldsymbol{S}$ and $\boldsymbol{S}'$ becomes very high for large N. To avoid precision loss when calculating the inverse, one then needs to increase the precision in the floating point operations. In MATLAB, we use MuPAD to handle this issue.

In the numerical experiments, we shall use this multipole expansion.

6.3 Far-Field Pattern

We solve the problem inside an artificial boundary, and thus computing the so-called near field. However, we are interested in the solution computed at a far field point $\boldsymbol{r}$. To solve this issue, one uses the integral solution given by (cf. [3, pp. 6-8])

$$p(\boldsymbol{r}) = \int_{\partial\Omega} \left[p(\boldsymbol{r}') \frac{\partial g(\boldsymbol{r}, \boldsymbol{r}')}{\partial n'} - g(\boldsymbol{r}, \boldsymbol{r}') \frac{\partial p(\boldsymbol{r}')}{\partial n'} \right] \mathrm{d}S(\boldsymbol{r}') \tag{6.16}$$

where $\boldsymbol{r}'$ a point on the surface $\partial\Omega$ and g is the free space Green's function defined (in 3D) by

$$g(\boldsymbol{r}, \boldsymbol{r}') = \frac{\mathrm{e}^{\mathrm{i}k|\boldsymbol{r}'-\boldsymbol{r}|}}{4\pi|\boldsymbol{r}'-\boldsymbol{r}|}.$$

We thus need to compute the derivative of both Green's function and the numerical solution for the pressure. We find

$$\frac{\partial g(\boldsymbol{r}, \boldsymbol{r}')}{\partial n'} = g(\boldsymbol{r}, \boldsymbol{r}') \frac{\mathrm{i}k|\boldsymbol{r}'-\boldsymbol{r}| - 1}{|\boldsymbol{r}'-\boldsymbol{r}|^2} (\boldsymbol{r}'-\boldsymbol{r}) \cdot \boldsymbol{n}'$$

where $\boldsymbol{n}'$ lies on Γ pointing "into" Ω^+. Ihlenburg defines the *far-field pattern* of a solution p by

$$F\left(\frac{\boldsymbol{r}}{r}\right) = r\mathrm{e}^{-\mathrm{i}kr} p(\boldsymbol{r}), \quad r \to \infty.$$

When plotting this field, we take the absolute value. Thus, $|F\left(\frac{\boldsymbol{r}}{r}\right)| = |rp(\boldsymbol{r})|$ is simply a scaling of the solution at a given radius r. Ihlenburg here uses an asymptotic approximation of the far field which is not at all necessary. This notation is only mentioned as we want to compare with the results of Ihlenburg.

When computing the far-field backscattered pressure with Equation (6.16), it is important to be aware of the potential source of error when the surface integral is approximated using Gaussian quadratures. To illustrate this effect, we integrate the exact solution at the boundary (using Gaussian quadratures) and compare the resulting backscattered pressure (computed from Equation (6.16)) with the exact solution at the far field point. In Figure 6.2 the relative absolute difference is plotted (using Equation (3.3)). Note

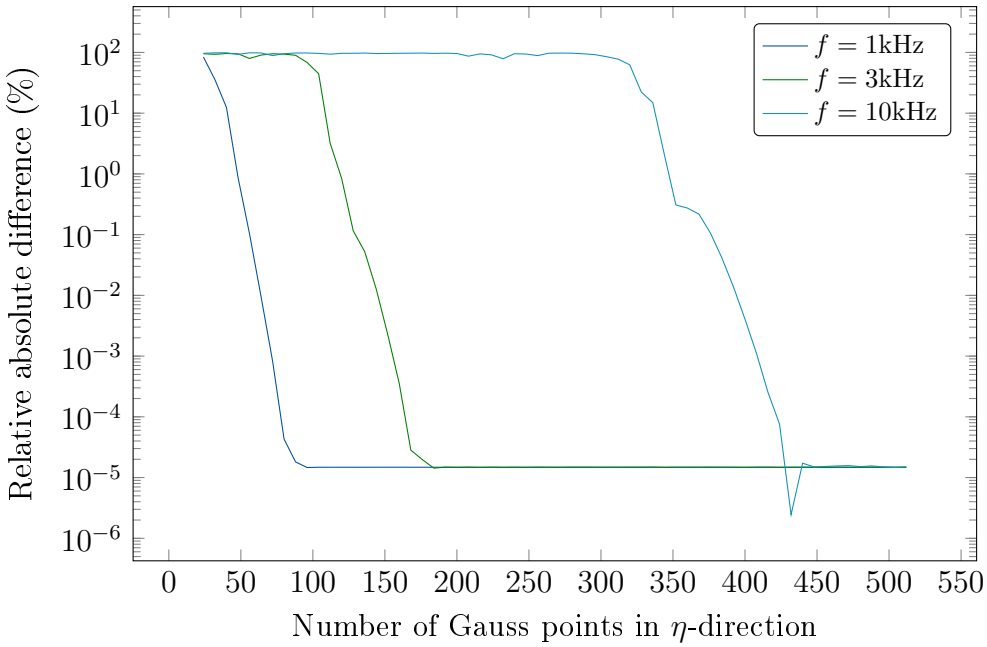


Figure 6.2: Scattering on rigid sphere: Error in computing the far field point due to Gaussian integration. The error is plotted against the number of Gauss points used in the η-direction. The total number of surface Gauss points is found by multiplying this number by 20. The refinement is only executed in the η-direction of the sphere (θ-direction) due to axisymmetry.

that due to symmetry, we only add new Gauss points in η-direction (where we start by inserting 6 knots). We do no refinement in the ξ-direction and will only use a total of 20 Gauss points in this direction. This is the reason why the error stabilizes at $10^{-5}\%$ since there will be an error in the integration for this direction. No order elevation is used. As we can see, we need to sample the solution with more points as the frequency increase. Recall that the frequency is given by

$$f = 2\pi\omega, \quad \text{where} \quad \omega = kc_{\text{f}}.$$

Hence, an increase in the frequency yields an increase in the wave number. If h is the distance between each evaluation point, one typically choose n_{res} evaluation points per wave length. That is,

$$n_{\text{res}} = \frac{\lambda}{h}$$

where

$$\lambda = \frac{2\pi}{k}$$

is the wavelength. As noted by Ihlenburg in [3], one typically choose $n_{\text{res}} = 10$ in practice. This rule applies to one dimensional analysis using linear interpolation functions. As we always use higher order functions at the surfaces, this rule will typically result in redundancy of evaluation points. The computation of the backscattered pressure does not serve as the most computationally expensive part, so to be on the safe side we shall use $n_{\text{res}} = 12$. We try to stay on the safe side here as no extensive analysis has been done in this regards.

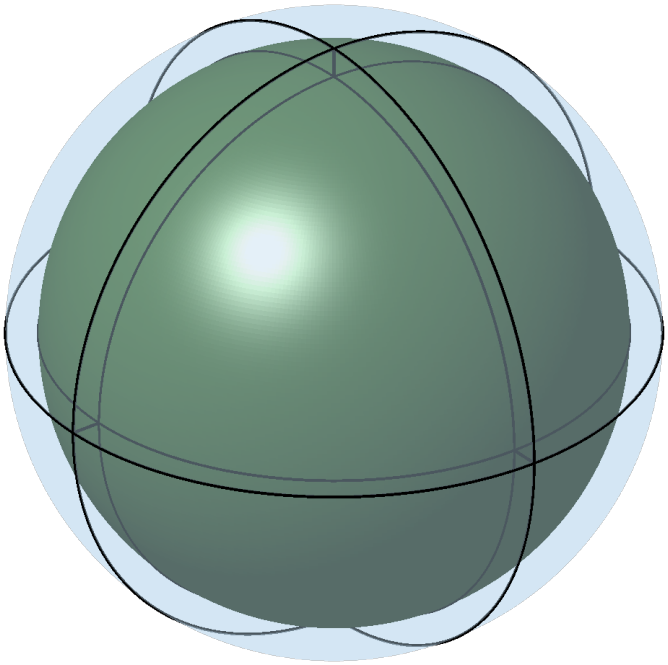


Figure 6.3: Scattering on a rigid sphere. The water is here parametrized with 8 elements (like a spherical shell).

6.4 Scattering on a rigid sphere

Consider a rigid sphere with radius $R = 5.075$ surrounded by water with parameters from Table 3.1. We shall consider scattering on this sphere from the incoming wave $p_{\text{inc}} = P_0 \mathrm{e}^{ikz}$, where we let the amplitude be $P_0 = 1$. In Section 3.4 the exact solution to this problem is presented. We shall use this example as the first test problem for the implementation of the infinite elements. We will then let the artificial boundary Γ_{a} be a sphere concentric with the rigid sphere with radius $R_{\text{a}} = 6$. And we shall use the BGU formulation with $N = 3$ for the infinite elements. The NURBS parametrization of this domain is depicted in Figure 6.3. Mesh 1 is then created by order elevation to $r = 2$ in ζ-direction, followed by the insertion of the knot $\zeta = 0.5$. Mesh 2 and 3 are then created by by adding 1 and 3 knots into the other parametric directions, respectively.

In Figure 6.4 we compare results from Mesh 1, 2 and 3 to the exact solution. What we observe is that the accuracy of the solution decays for larger wave numbers. This is known as the *pollution effect* and is discussed in [34]. It is there proven that it is impossible to eliminate this pollution effect using FEM in two and more space dimensions. This is the reason why FEM is not so suited for higher frequencies. To obtain accurate solutions for higher wave numbers, more degrees of freedoms must be added, which in turn increases the computational time.

6.5 Scattering on BeTSSi model 3

We shall now consider incoming waves from different directions scattered by BeTSSi Model 3. The original model is aligned with the x-axis, but we shall here without loss of generality align the model with the z-axis. This is convenient as the prolate spheroidal

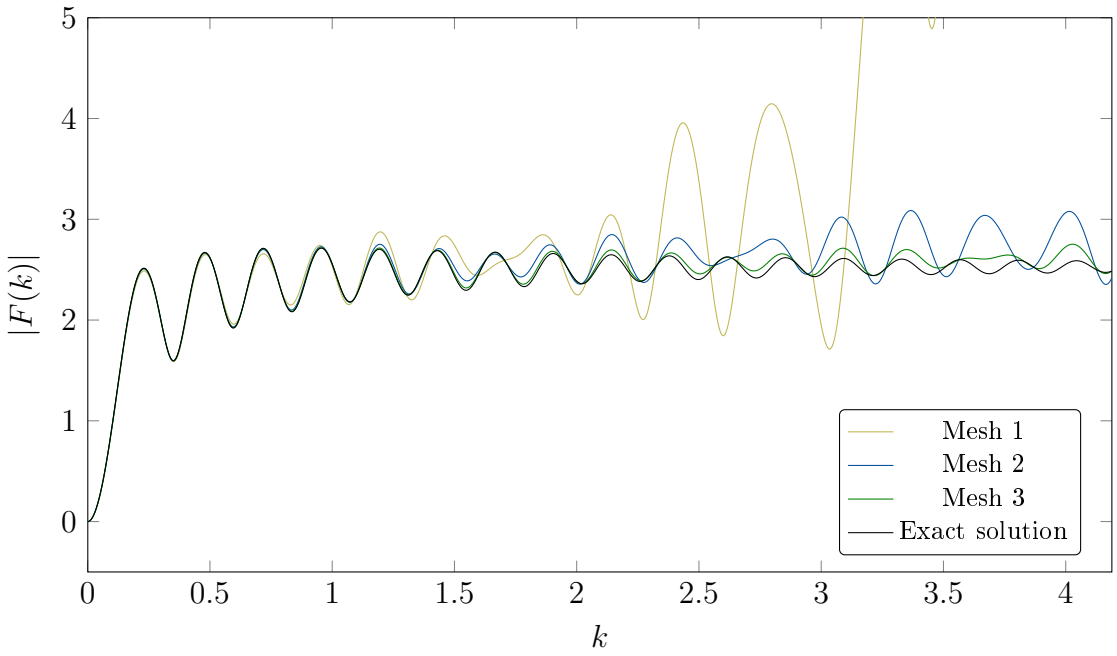


Figure 6.4: Scattering on a rigid sphere: Convergence results.

coordinate system (used for infinite elements) is also aligned with the z-axis. In addition to this rotation, it is also convenient to enable translation of the model. That is, we want to translate the origin of the model to a point $\boldsymbol{x}_0$.

If a plane wave is traveling in the direction $\boldsymbol{k}$, then it can be written in the following form

$$p_{\text{inc}} = P_0 \mathrm{e}^{\mathrm{i}\boldsymbol{k}\cdot(\boldsymbol{x}-\boldsymbol{x}_0)}$$

Once again, in the numerical experiments we shall always use $P_0 = 1$.

We shall consider the receiver of this wave to be located at

$$\boldsymbol{P}_{\text{far}} = r_{\text{f}}(\cos\beta_{\text{f}}\sin\alpha_{\text{f}}, \sin\beta_{\text{f}}, \cos\beta_{\text{f}}\alpha_{\text{f}}) + \boldsymbol{x}_0$$

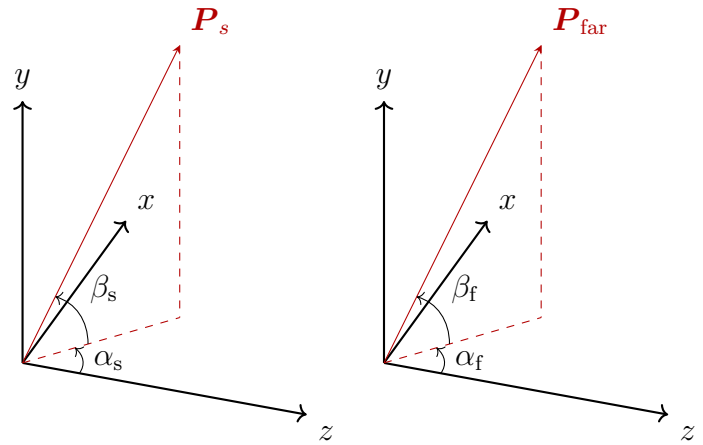


Figure 6.5: Scattering on BeTSSi model 3: Aspect angle α and elevation angle β for BeTSSi model 3 which is aligned with the z-axis. Note that $\boldsymbol{x}_0 = \boldsymbol{0}$.

where α_s and β_s is aspect angle and the elevation angle, respectively (cf. Figure 6.5). A plane wave originated from the source point (r_s, α_s, β_s) point (and traveling towards $\boldsymbol{x}_0$), thus have the wave number

$$\boldsymbol{k} = -k\cos\beta_s \sin\alpha_s \boldsymbol{e}_x - k\sin\beta_s \boldsymbol{e}_y - k\cos\beta_s \sin\alpha_s \boldsymbol{e}_z.$$

The normal derivative on the surface of any smooth geometry may be computed by

$$\partial_n p_{\text{inc}} = \boldsymbol{n} \cdot \nabla p_{\text{inc}} = \mathrm{i}\boldsymbol{k} \cdot \boldsymbol{n} p_{\text{inc}}$$

and the target strength (TS) is calculated by

$$TS = 20\log_{10}\left(\|\boldsymbol{P}_{\text{far}}\| \frac{|p(\boldsymbol{P}_{\text{far}})|}{P_0}\right).$$

6.5.1 Constructing the mesh

The mesh is constructed by joining the NURBS surface of BeTSSi model 3 with a NURBS surface of a prolate spheroid. The prolate spheroid is just a special example of the more general ellipsoid which may exactly be parameterized by a NURBS surface (cf. Appendix D.6). Both these surfaces has been parametrized with the same knot vector Ξ (in the azimuthal direction). However, the parameter direction in the polar direction (η-direction) has different knot vectors for the two surfaces. To join these two surfaces into a full 3D NURBS solid, one needs to insert the missing knots in order to create identical knot vectors for the surfaces. The knot vectors in the η-direction for the two surfaces are

$$\mathcal{H}_{\text{M3}} = \{0, 0, 0, 0.265, 0.265, 0.8, 0.8, 1, 1, 1\}$$

and

$$\mathcal{H}_{\text{spheroid}} = \{0, 0, 0, 0.5, 0.5, 1, 1, 1\}$$

for the surface of BeTSSi model 3 and the prolate spheroid, respectively. Thus, our initial mesh has the knot vector

$$\mathcal{H} = \{0, 0, 0, \eta_1, \eta_1, \eta_2, \eta_2, \eta_3, \eta_3, 1, 1, 1\}$$

with $\eta_1 = 0.265$, $\eta_2 = 0.5$ and $\eta_3 = 0.8$. The use of repetitive knots is essential for BeTSSi model 3 as there is C^0-continuity in the exact geometry at these knots.

The parametrization in the radial direction (ζ-direction) adds the first degree of freedom in construction the mesh. One could simply use a linear parametrization in this direction. However, we shall choose a second degree NURBS parametrization for the ζ-direction. We then need to insert an extra layer of control point between the two existing surfaces. We shall use a set of control points which resembles the control points at the surfaces of BeTSSi model 3 in such a way that the mesh lines in the ζ-direction is perpendicular at the hemispherical ends (due to the lacking smoothness of BeTSSi model 3 it is not possible to have perpendicularity everywhere). The resulting 3D mesh is depicted in Figure 6.6. In Figure 6.7a we show the corresponding cross-section of the mesh in the upper xz-plane. In the azimuthal direction (ξ-direction), we use uniform refinement. It is not necessarily the best to use uniform refinement in the other two directions. The result of purely uniform refinement is depicted in Figure 6.7b. This results in areas where large

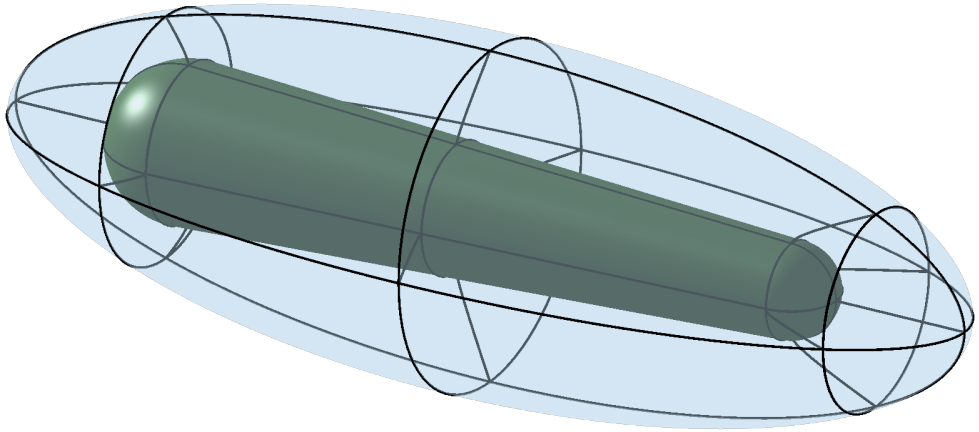
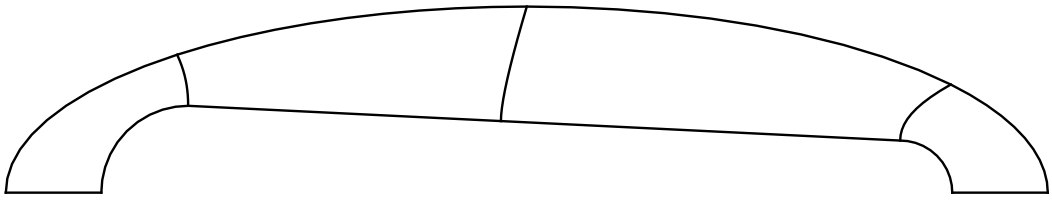
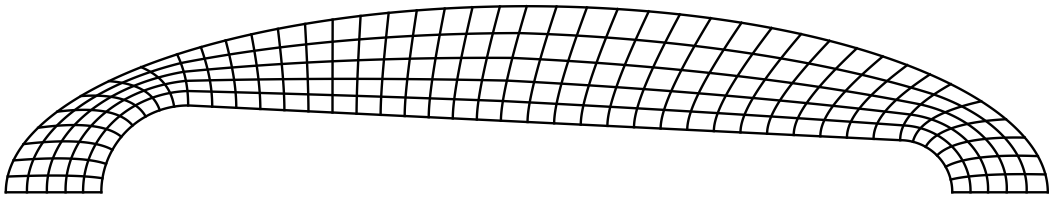


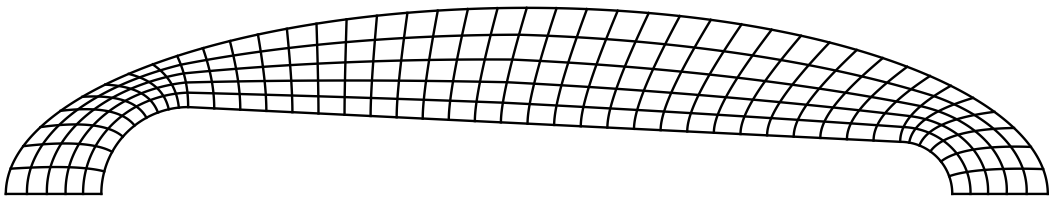
Figure 6.6: Rigid scattering on BeTSSi model 3: The initial mesh of the fluid surrounding BeTSSi model 3.



(a) Mesh 0



(b) Mesh 1 uniform parametrization



(c) Mesh 1 non uniform parametrization

Figure 6.7: Rigid scattering on BeTSSi model 3: Meshes of fluid, in the upper xz-plane. The full 3D mesh is obtained by rotating this mesh about the major axis (z-axis).

elements occur. This is typically not wanted, and we shall therefore use non uniform refinement resulting in the mesh in Figure 6.7c. This mesh has been constructed in such a way that we have uniform length of the elements along the polar direction at the prolate spheroid. In the following we shall describe this process in detail where we try to find the optimal mesh with respect to a uniform distribution of elements.

We only need to consider inserting knots in the NURBS curve where $\zeta = 1$ and $\xi = 0$ (this will be the upper edge in Figure 6.7a) due to symmetry. Find first the physical location of the knots η_1, η_2 and η_3 such that the corresponding polar angles θ_1, θ_2 and θ_3 in the prolate spheroidal coordinate system may be calculated. Calculate the arc length between these angles with the following formula

$$L(a,b) = \int_a^b \sqrt{\left(\frac{\partial x}{\partial \theta}\right)^2 + \left(\frac{\partial y}{\partial \theta}\right)^2 + \left(\frac{\partial z}{\partial \theta}\right)^2}\,\mathrm{d}\theta = \int_a^b \sqrt{r_{\mathrm{a}}^2 - f^2\cos^2\theta}\,\mathrm{d}\theta.$$

which may not be written out in closed form (it may be written out in terms of the elliptic integral, but we shall simply use numerical integration for evaluations). If s is the mesh number, we shall insert

$$m(a,b) = \lceil 0.55 s L(a,b) \rceil$$

knots between the pair of angles $(0,\theta_3)$, (θ_3,θ_2), (θ_2,θ_1) and (θ_1,π). For mesh 1 ($s=1$) this implies a total of 39 new knots inserted in the η-direction (correspondingly, 16 knots have been inserted in the ξ-direction and 4 knots in the ζ-direction). This results in a almost uniform partition of the curve. A perfect partition of the curve imposes constraints on the knots η_1 and η_3, but we want to choose these knots in BeTSSi model 3 in order to obtain the optimal mesh. Consider the insertion of $m(\theta_3,\pi)$ knots between the angles θ_3 and π (the larger hemisphere). We start with $\theta_{\mathrm{prev}} = \theta_3$, and looping through the $m(\theta_3,\pi)$ angles by finding the next angle θ such that

$$L(\theta_{\mathrm{prev}},\theta) = \frac{L(\theta_3,\pi)}{m(\theta_3,\pi)+1}.$$

We find this angle using the bisection method. We may now evaluate the Cartesian coordinate $[x,y,z]^\top$ of this angle, and use the built in MATLAB function `fminsearchbnd` to find

$$\eta = \min_{\eta\in[0,1]} \left\| [x,y,z]^\top - \boldsymbol{X}(0,\eta,1) \right\|.$$

It is possible to optimize this last step by using Newton iteration as mentioned [1, p. 4191]. A corresponding procedure is done for the other intervals. The advantage of this new mesh using non uniform refinement is that we now get small areas in which we get "too many" elements in contrast to areas in which we get too few elements. Finally, consider the refinement in the ζ-direction. As the intermediate layer of control points between the surface of BeTSSi model 3 and the prolate spheroid is placed close to the surface of BeTSSi model 3, a uniform refinement in $\mathcal{Z}$ will result in a decreasing density of elements from this surface. But we want the elements to form a uniform density of elements throughout the mesh, so we employ non uniform mesh refinement also in this direction. In Figure 6.9 we plot the mesh for $s = 2,3,4,5$. It should be noted that uniform refinement in ξ-direction does not imply uniform refinement in the physical azimuthal direction (ϕ-direction). However, as we can see in Figure 6.8, this is not far from being the case.

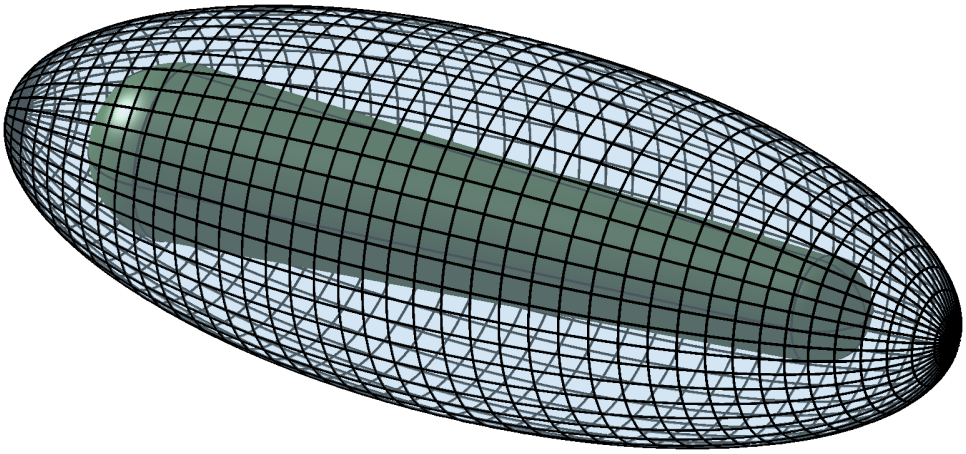
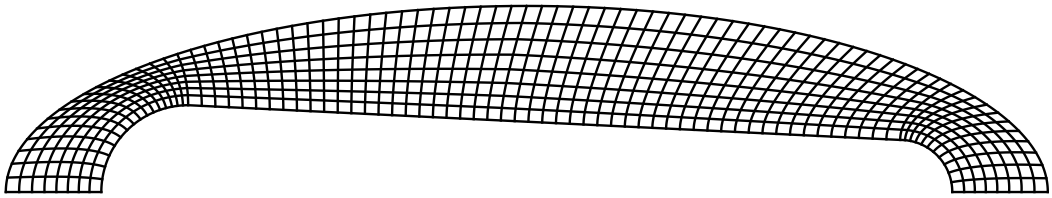
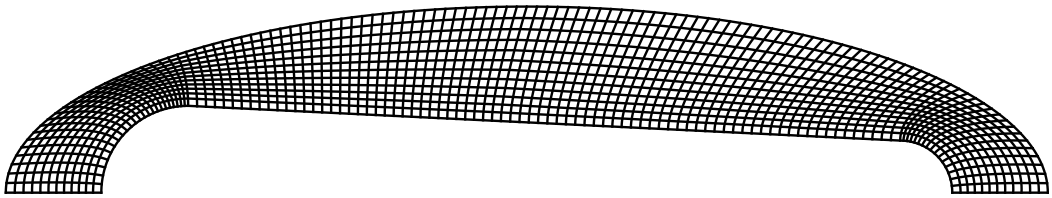


Figure 6.8: Rigid scattering on BeTSSi model 3: Visualization of the surface mesh for the prolate spheroid surrounding BeTSSi model 3 corresponding to mesh 1.

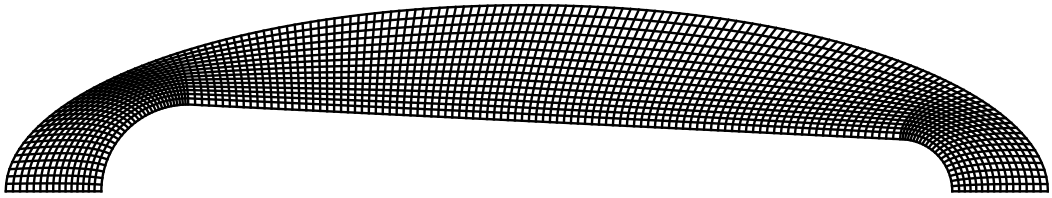
This refinement procedure could obviously have been improved if local refinement was possible. This could be done by T-splines (cf. [21]) or LR B-splines (cf. [20]), but this is far beyond the scope of this thesis.



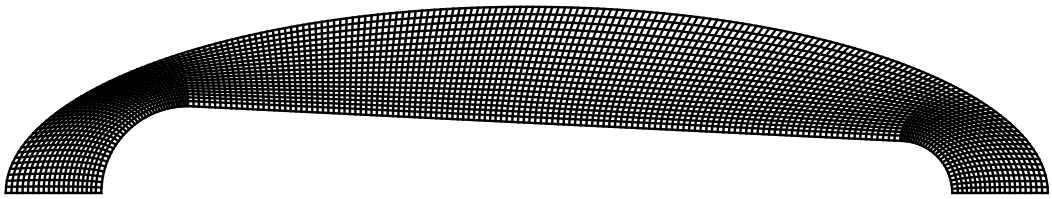
(a) Mesh 2



(b) Mesh 3



(c) Mesh 4



(d) Mesh 5

Figure 6.9: Rigid scattering on BeTSSi model 3: Meshes of fluid, in the upper xz-plane. The full 3D mesh is obtained by rotating this mesh about the major axis (z-axis).

Chapter 7

Fluid-structure interaction

So far we have considered linear elasticity problems and exterior Helmholtz problems separate. We shall now combine these two problems into a fluid-structure interaction problem (FSI).

In [3, pp. 13-14] Ihlenburg briefly derives the governing equations. We repeat them here using similar notations as Ihlenburg:

$$\sigma_{ij,j} + \omega^2 \rho_s u_i = 0 \qquad \text{in} \quad \Omega \tag{7.1}$$

$$\rho_f \omega^2 u_i n_i - \frac{\partial p}{\partial n} = \frac{\partial p_{\text{inc}}}{\partial n} \qquad \text{on} \quad \Gamma \tag{7.2}$$

$$\sigma_{ij} n_i n_j + p = -p_{\text{inc}} \qquad \text{on} \quad \Gamma \tag{7.3}$$

$$\Delta p + k^2 p = 0 \qquad \text{in} \quad \Omega^+, \tag{7.4}$$

$$\frac{\partial p}{\partial R} - \mathrm{i}kp = o(R^{-1}) \qquad R \to \infty, \tag{7.5}$$

For the domain Ω we have shown that we obtain the weak formulation (see Equation (4.13) with no body forces)

$$\int_\Omega v_{i,j}\sigma_{ij} - \rho_s \omega^2 u_i \bar{v}_i \, \mathrm{d}\Omega = \int_\Gamma v_i(\sigma_{ij} n_j) \, \mathrm{d}\Gamma. \tag{7.6}$$

The integrand on the right hand side may be rewritten using Equation (7.3) in the following way. Let h_i be the components (in Cartesian coordinates) of the exterior traction vector on Γ. That is to say,

$$h_i = \sigma_{ij} n_j.$$

Consider a point P on Γ with normal vector $\boldsymbol{n}$. We may then create a local orthogonal coordinate system at this point with unit vectors $\boldsymbol{e}_n$, $\boldsymbol{e}_{t_1}$ and $\boldsymbol{e}_{t_2}$ where the latter two vectors represents the tangential basis vectors to the surface Γ.

As the scalar product is invariant to orthogonal transformations we get

$$h_i v_i = h_x v_x + h_y v_y + h_z v_z = h_n v_n + h_{t_1} v_{t_1} + h_{t_2} v_{t_2}.$$

The pressure from the fluid only exerts forces normal to the surface Γ. The static equilibrium condition for the traction at P is then

$$h_{t_1} = 0, \qquad h_{t_1} = 0, \quad \text{and} \quad h_n = -(p_{\text{inc}} + p).$$

The scalar product may therefore be written as

$$h_i v_i = -(p_{\text{inc}} + p)v_n = -(p_{\text{inc}} + p)v_i n_i.$$

Equation (7.6) thus reduces to

$$\int_\Omega v_{i,j}\sigma_{ij} - \rho_s \omega^2 u_i v_i \,\mathrm{d}\Omega = -\int_\Gamma (p_{\text{inc}} + p) v_i n_i \,\mathrm{d}\Gamma. \tag{7.7}$$

Moreover, from Equation (6.1)

$$\int_{\Omega^+} \nabla q \cdot \nabla p - k^2 q p \,\mathrm{d}\Omega = -\int_\Gamma q \frac{\partial p}{\partial n} \,\mathrm{d}\Gamma$$

where we have changed sign of the right hand side in order to get a normal vector that points outwards. Using now Equation (7.2) we get

$$\int_{\Omega^+} \nabla q \cdot \nabla p - k^2 q p \,\mathrm{d}\Omega = -\int_\Gamma q \left(\rho_f \omega^2 u_i n_i - \frac{\partial p_{\text{inc}}}{\partial n}\right) \mathrm{d}\Gamma. \tag{7.8}$$

Scaling Equation (7.7) with the scaling factor $\rho_f \omega^2$ (for symmetry) we may add this equation to Equation (7.8) to obtain

$$\begin{aligned} \int_{\Omega^+} \nabla q \cdot \nabla p - k^2 q p \,\mathrm{d}\Omega + \rho_f \omega^2 \left(\int_\Gamma q u_i n_i \,\mathrm{d}\Gamma + \int_\Gamma v_i n_i p \,\mathrm{d}\Gamma\right) \\ + \rho_f \omega^2 \int_\Omega v_{i,j}\sigma_{ij} - \rho_s \omega^2 u_i v_i \,\mathrm{d}\Omega \\ = -\rho_f \omega^2 \int_\Gamma p_{\text{inc}} v_i n_i \,\mathrm{d}\Gamma + \int_\Gamma \frac{\partial p_{\text{inc}}}{\partial n} q \,\mathrm{d}\Gamma \end{aligned} \tag{7.9}$$

where $\boldsymbol{n} = \{n_1, n_2, n_3\}$ points outwards from the solid. Ihlenburg now continues by defining

$$\begin{aligned} \mathcal{B}(\mathcal{U}, \mathcal{V}) = \int_{\Omega^+} \nabla q \cdot \nabla p - k^2 q p \,\mathrm{d}\Omega + \rho_f \omega^2 \left(\int_\Gamma q u_i n_i \,\mathrm{d}\Gamma + \int_\Gamma v_i n_i p \,\mathrm{d}\Gamma\right) \\ + \rho_f \omega^2 \int_\Omega v_{i,j}\sigma_{ij} - \rho_s \omega^2 u_i v_i \,\mathrm{d}\Omega \end{aligned}$$

and

$$(\mathcal{F}, \mathcal{V}) = -\rho_f \omega^2 \int_\Gamma p_{\text{inc}} v_i n_i \,\mathrm{d}\Gamma + \int_\Gamma \frac{\partial p_{\text{inc}}}{\partial n} q \,\mathrm{d}\Gamma$$

where $\mathcal{U} = (\boldsymbol{u}, p)$ and $\mathcal{V} = (\boldsymbol{v}, q)$. Moreover, define the Sobolov spaces $\mathcal{H} = H^1(\Omega_s) \times H^{1+}_w(\Omega^+)$ and $\mathcal{H}^* = H^1(\Omega_s) \times H^1_{w^*}(\Omega^+)$. The weak formulation for the FSI problem then becomes: Find all $\mathcal{U} \in \mathcal{H}$ such that

$$\mathcal{B}(\mathcal{U}, \mathcal{V}) = (\mathcal{F}, \mathcal{V}), \quad \forall \mathcal{V} \in \mathcal{H}^*.$$

Chapter 8

Computational aspects

In Figure 8.1 we give an overview of the code for the full fluid structure interaction program for both monostatic and bistatic cases. The only part which is not parallelized is the initialization part of the code. The time elapsed by this part is negligible in comparison with any other part of the program, so any effort for optimizing this part was omitted. In the following we shall briefly present the main content of each step.

8.1 Initialization

The initialization part consist of defining all relevant parameters and constructing the mesh for analysis. To create the IGA mesh, we first import data for the coarsest mesh of the NURBS object (i.e. data set's found in Appendix D). One typically first constructs the NURBS object for the solid domain as this domain should be the exact representation of the given object. In contrast, we have more degrees of freedom when construction the fluid mesh. The mesh for the fluid domain must be constructed by finding a suitable spheroid which may be wrapped closely around the obstacle without creating to much of a distortion in the resulting mesh (see Subsection 6.5.1). Although it may certainly be possible, it has not been implemented an automated algorithm for finding such a spheroid. So in the analysis, we have found the fluid mesh by trial and failure. When the two meshes have been constructed (two NURBS patches) one has many opportunities of k-refinement. The two algorithms `elevateNURBSdegree` and `insertKnotsInNURBS` have been implemented for this purpose (see Subsection 4.2.2). Typically we order elevate first to the desired degrees in ξ-, η- and ζ-direction. Then we insert the desired number of knots in the desired locations (creating more elements for analysis). From this point, we are able to create the IGA mesh. We first construct an ordering of the available NURBS basis functions, then we construct a connectivity array which for each element links every basis functions having support on this element. From here, it is important to have knowledge if there is any periodicity involved for the NURBS patch. If this is the case, one must find the overlapping control points causing the periodicity. There could potentially be more than two control points at the same physical location. In any case, define the control point with lowest index to be the "parent function" and the rest "children functions". Then one constructs an array mapping all children functions to the parent function. This array is then used to modify the mentioned connectivity array. In

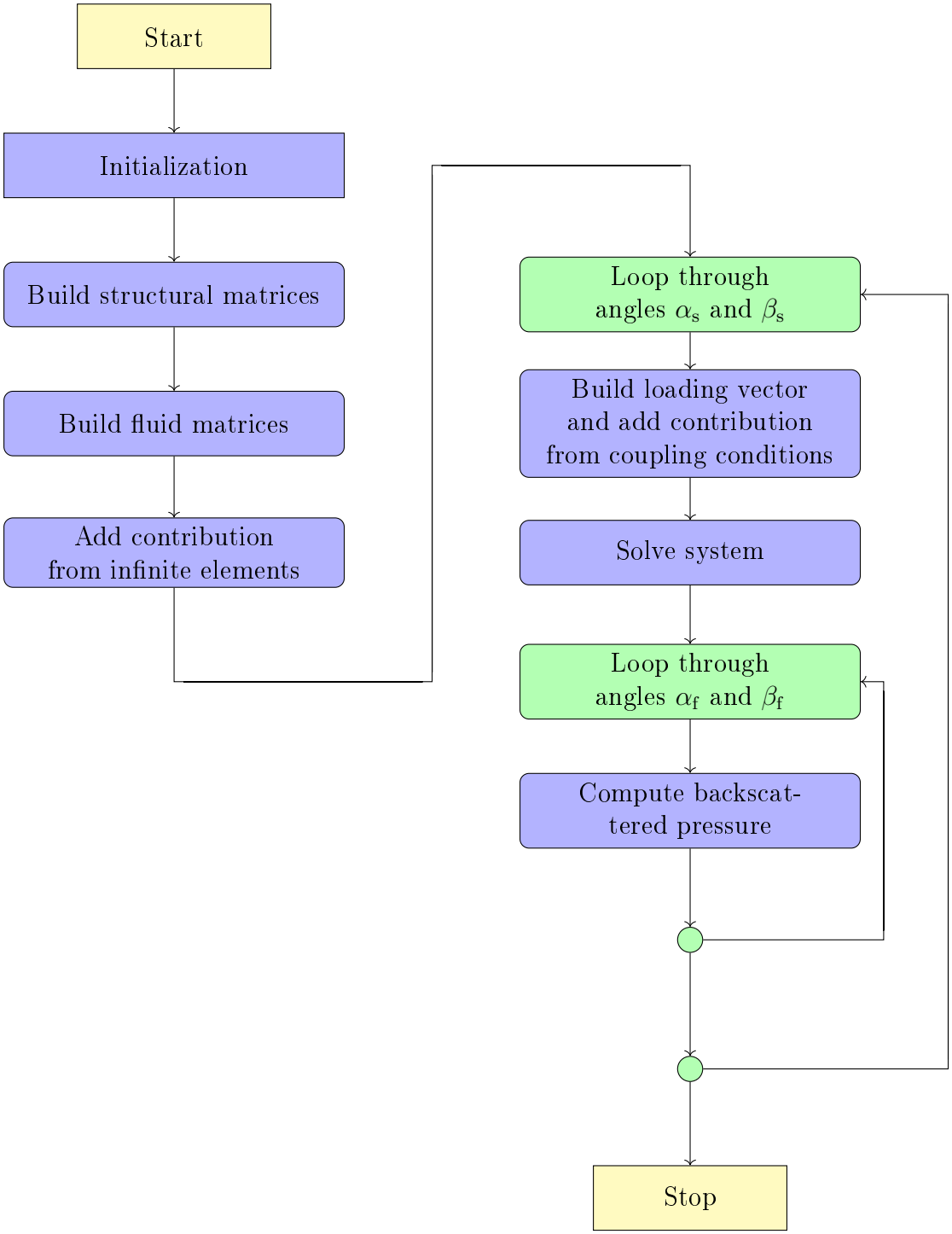


Figure 8.1: The FSI program: The flowchart of the full FSI program analyzing both monostatic and bistatic cases. The boxes with rounded corners are parallelized.

addition to keeping track of the indices for the parent functions, one must also store the indices for the children functions as these degrees of freedom must be removed from the global system matrix.

8.2 Structural matrices and fluid matrices

When constructing the stiffness matrix and the mass matrix, we create four arrays for these two matrices. The first two arrays will contain the indices (columns and rows) in the global matrix. The other two arrays stores the contribution at these indices for each of the two matrices. We continue by creating an array of Gaussian quadrature points and weights. The code supports 64 Gaussian quadrature points per direction such that one is able to evaluate a 3D IGA element at $64^3 = 262144$ points. However, one typically determines the number of control points by the NURBS degree for each of the parameter directions. More specifically, if the degree in ξ direction is p, then we use $p+1$ quadrature points in this direction. Thus, we have $(p+1)(q+1)(r+1)$ quadrature points per volume element.

It is very important to note that the global stiffness matrix is very sparse. Not only is it a huge advantage in MATLAB to have the matrix in sparse format when solving the linear system, but also when assembling the matrix. If the matrix is made sparse only after assembly, then the initialization would require MATLAB to allocated place for a full stiffness matrix. This is very memory consuming and should be avoided if one wants to run the program with many degrees of freedom. A matrix in sparse format contains 3 columns; the first two represent the indices in the matrix and the third column represent the corresponding value. The problem of preallocation may be solved by first construct these three columns in three arrays. By noting that each element stiffness matrix has $[3(p+1)(q+1)(r+1)]^2$ number of components, we initialize the arrays by the following.

```
n_en = (p+1)*(q+1)*(r+1);
sizeOfk_e = (3*n_en)^2;
indices = zeros(3*n_en,noElems);
spIdxRow = zeros(sizeOfk_e,noElems);
spIdxCol = zeros(sizeOfk_e,noElems);
Kvalues = zeros(sizeOfk_e,noElems);
```

Note that we initialize these "vectors" in matrix form as this allows us to easily parallelize using `parfor` in MATLAB.

Note that we do not here sum overlapping element matrices, but this is automatically done when the `sparse` function is called in MATLAB.

```
K = sparse(spIdxRow,spIdxCol,values);
```

That is, there will be index combinations which will repeat and therefore this method would require a lot more memory. Using server with 256GB memory space, one is able to run this implementation with up to 10^6 degrees of freedom, which is more than enough for this thesis. Thus, the results evaluated for this thesis is done relatively fast.

The other obvious method is simply using

```
K(sctrB,sctrB) = K(sctrB,sctrB) + k_e;
```

which typically results in the MATLAB warning "This sparse indexing expression is likely to be slow". In [35] it is described how one could optimize the assembly, and in [36] we find theory around efficiently use of quadrature for NURBS. In [37] a new stack based representation of the sparse matrix is presented, which could speed up the matrix assembly part. This is however outside the scope of this thesis, and is therefore not included.

We are now ready to loop through all structure elements and calculate their contribution to the stiffness and mass matrix. For each element in this loop (which is parallelized) we must loop through all quadrature points. For each quadrature point in this inner loop, we evaluate all the NURBS basis functions which have support on the current element. The Jacobian is then computed (by Equation (4.20)) in order to find the derivative of the basis functions with respect to physical coordinates (Equation (4.22)) which is then used to find the strain matrix $\boldsymbol{B}$ (Equation (4.18)). Note that if body forces is to be applied to the solid, these will be calculated in this same loop. So one must not mistake the part of building the loading vector with the implementation of body forces. However, we never encounter body forces in the analysis of acoustic scattering, but it has been implemented anyways as some static elasticity problems involving body forces has been analyzed in this thesis (the Kneaded cylinder in Subsection 4.6.2 and Scordelis-Lo Roof in Appendix C.1.1).

After looping through all elements, one creates the sparse matrices $\boldsymbol{K}$ and $\boldsymbol{M}$ using the `sparse` function in MATLAB. These matrices are then added (with a scaling factor for the mass matrix). The reason why they are not added during calculation is simply to create a more general function which handles the elastostatic case as well (no construction of the mass matrix).

The next step is to construct the matrices for the fluid part of the domain. The setup is almost identical to the case for the structure matrices. However, we are now working with scalar valued domain instead of a vector valued domain. This will affect the indexing. In addition, the bilinear form is different for the two cases. After computing the fluid matrices, the result is added to the global matrix.

8.3 Contribution from infinite elements

The infinite elements are implemented using the multi-pole expansion after Burnett in [10]. We generalize this work, which was based on the BGU formulation, to the other three formulations BGC, PGU and PGC (see Section 6.2).

One first needs to locate the indices for the NURBS basis functions having support on the artificial boundary Γ_a. In addition, the radial integrals in the bilinear or sesquilinear form must be computed (depending on which infinite element formulation is chosen). Finally, one must compute the coefficients used for the radial basis functions in the infinite elements (from the relation Equation (6.15)). These two last step suffers from round off errors for a high number of infinite element radial basis functions N. The way to solve this is to increase the precision using the mentioned MuPAD in MATLAB.

We are now ready to loop through all surface elements on Γ_a. Once again, for each element, we have an inner loop over all quadrature points. And for each quadrature points, the NURBS basis functions are evaluated, such that the Jacobi matrix can be computed. This is in turn used to compute the derivative with respect to the prolate spheroidal coordinates. In this regards it has been implemented a function which evaluates (r, θ, ϕ) for a given point (x, y, z) (using the relations in Equation (B.32)). Moreover, the derivatives of (r, θ, ϕ) with respect to (x, y, z) must be computed (using the relations in Equation (B.34)). We may then compute the derivatives of (r, θ, ϕ) w.r.t, the surface parameters ξ and η using Equation (6.12). These derivatives are finally used to evaluate the derivatives of the basis functions with respect to the angles θ and ϕ using Equation (6.11). This enables us to compute the contributions to the surface integrals $J_{AB}^{(1)}$, $J_{AB}^{(2)}$, $J_{AB}^{(3)}$, $J_{AB}^{(4)}$ and $J_{AB}^{(5)}$ (expressions in Equation (6.7)). Note that we do not compute the full bilinear or sesquilinear form in this inner loop, as the radial integrals is independent of the angular coordinates (to avoid redundant computations). After looping through the quadrature points, we continue the loop on the current element and evaluates the bilinear or the sesquilinear form. This is done by looping through all radial basis functions in the infinite elements. The result is then collected and added to the global matrix.

8.4 Loading vector and coupling condition

The coupling conditions connects the two spaces for the solid domain and the fluid domain. This involves a coupling of two NURBS objects. The implementations assume the surfaces to be connected have the same NURBS parametrization. It also assumes the NURBS patches to be parametrized in such a way that these surfaces are given at $\zeta = 0$ for the fluid and $\zeta = 1$ for the solid. This causes restriction in the refinement process; if we want to, say, insert a knot in the ξ direction, or order elevate in this direction in the fluid, we must do the same refinement process in the solid.

Keeping these remarks in mind, we start again by constructing the surface mesh. This mesh is constructed on the basis of the surface at $\zeta = 1$ for the solid. We then loop through these surface elements, and for each element we must loop through the quadrature points. For each such quadrature point we now evaluate 2D NURBS functions, as we know the 3D NURBS functions reduce to 2D NURBS functions when ζ is constant. This obviously saves a lot of computations. Once again, these evaluations are used to compute the (2D) Jacobian which is in this case not used to evaluate derivatives of the NURBS functions w.r.t. physical coordinates, as they do not appear in the coupling conditions. The Jacobian is only used to compute the cross product needed to compute the surface integral (see Equation (4.23)). This cross product is also needed to determine the normal vector in the coupling term in the FSI bilinear form (Equation (7.9)). As we want this normal vector to point out from the solid domain, and we consider the NURBS parametrization of the solid domain, the normal vector is given by

$$\boldsymbol{n} = \frac{\frac{\partial \boldsymbol{X}}{\partial \xi} \times \frac{\partial \boldsymbol{X}}{\partial \eta}}{\left\| \frac{\partial \boldsymbol{X}}{\partial \xi} \times \frac{\partial \boldsymbol{X}}{\partial \eta} \right\|}$$

for right hand oriented NURBS parametrization. It is here assumed that the parametrization is orientable. The parametrization is right hand oriented whenever $\det(\boldsymbol{J}) > 0$

throughout the domain. If $\det(\boldsymbol{J}) < 0$ throughout the domain, we simply need to change the sign of the normal vector in the expression above.

We continue by computing the physical coordinates in order to evaluate the given incident field p_{inc} and its derivative. The result is then collected and added to the global matrix. Note that the other contributions to the global matrix is independent of the incident field such that they only need to be computed once (thus we store this part). Moreover, if only hard walled boundary conditions are considered (only fluid, and no coupling), then there is no update in the global matrix in this loop. If a direct solver is chosen, one may split the loop over the angles α_{s} and β_{s} such that the factorization is only done once.

8.5 Solving the system

If there are many angles to be analyzed, the loops over these angles should be parallelized. However, if one only consider few angles (for instance the bistatic case when only one pair of α_{s} and β_{s} is considered), one should take the parallelization over the loop which runs through the surface elements.

When a direct solver is to be used, the backslash operator in MATLAB will elegantly solve the problem in a efficient way. However, if the LU factorization may be reused (which is the case when no coupling is present) one should use the build in `lu` function. If the LU factorization is reused, one should no longer parallelize over the angles as this would require the LU factorization to be copied in each thread (a wast amount of extra memory is thus acquired). More details on the subject can be found in [38].

For a given incident field, consider now the full global matrix. We should then remove the degrees of freedom corresponding to all children functions. After this is done, we may now investigate the resulting structure of the global matrix A.

In Figure 8.2 we show the results of typing `spy(A)` in MATLAB (with additional effects added). The function gives an overview of the location of the nonzero entries in the global matrix. This result comes from elastic scattering on a spherical shell, where the mesh is simply a union of two spherical shells; one for the solid and the other for the fluid. In particular, the polynomial degree for the solid is $p_{\text{s}} = q_{\text{s}} = r_{\text{s}} = 2$ and for the fluid $p_{\text{f}} = q_{\text{f}} = 2$ and $r_{\text{f}} = 1$. No refinement is done on the coarsest mesh (cf. Figure 6.3), except for two knots in the radial direction for both the solid and the fluid (namely $\zeta = \frac{1}{3}$ and $\zeta = \frac{2}{3}$). That is, the number of basis functions in the solid are $n_{\text{s}} = 9$, $m_{\text{s}} = 5$ and $l_{\text{s}} = 5$ (originally 3) and $n_{\text{f}} = 9$, $m_{\text{f}} = 5$ and $l_{\text{s}} = 4$ (originally 3) for the fluid (cf, Appendix D.5). Due to periodicity, some of these basis functions should be considered as the same. For a given set of basis functions in the radial direction we get $n_{\text{s}}m_{\text{s}} = 45$ basis functions in the solid (and correspondingly in the fluid). Due to gluing, this number reduces to 26. The number represent the width of the "squares" showing up all over the matrix. Obviously we get $26l_{\text{s}} = 130$ basis functions for each of the spatial components of the solid, such that there is in total 390 degrees of freedom in the solid. Moreover, we see $26l_{\text{f}} = 104$ basis functions for the finite element (FE) part of the fluid. As we use $N = 3$ basis functions in the radial direction for the infinite elements, we can observe an additional $26(N-1) = 52$ basis functions corresponding to the infinite elements (IE). The total number of degrees of freedom for this mesh is thus $390 + 104 + 52 = 546$. Also

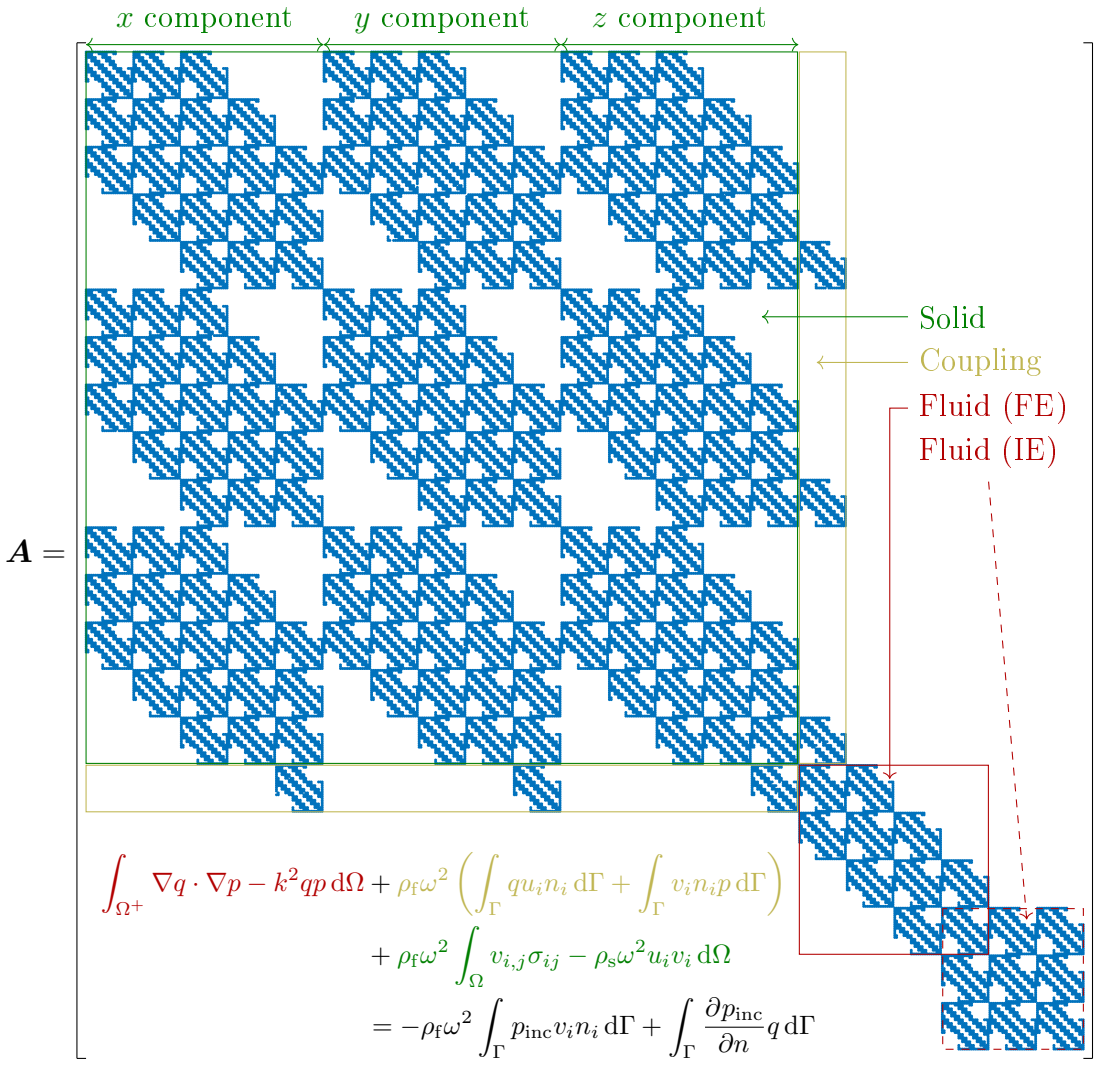


Figure 8.2: Elastic scattering on spherical shell: Location of non zero entries in the sparse global matrix A. The polynomial degree is 2 in every parameter directions for both the solid and the fluid except in the radial direction for the fluid (where it is 1). The number of radial basis functions for the infinite elements is $N = 3$. The only h-refinement has been in the radial direction; there has been inserted 2 knots in the radial direction for both the solid and the fluid.

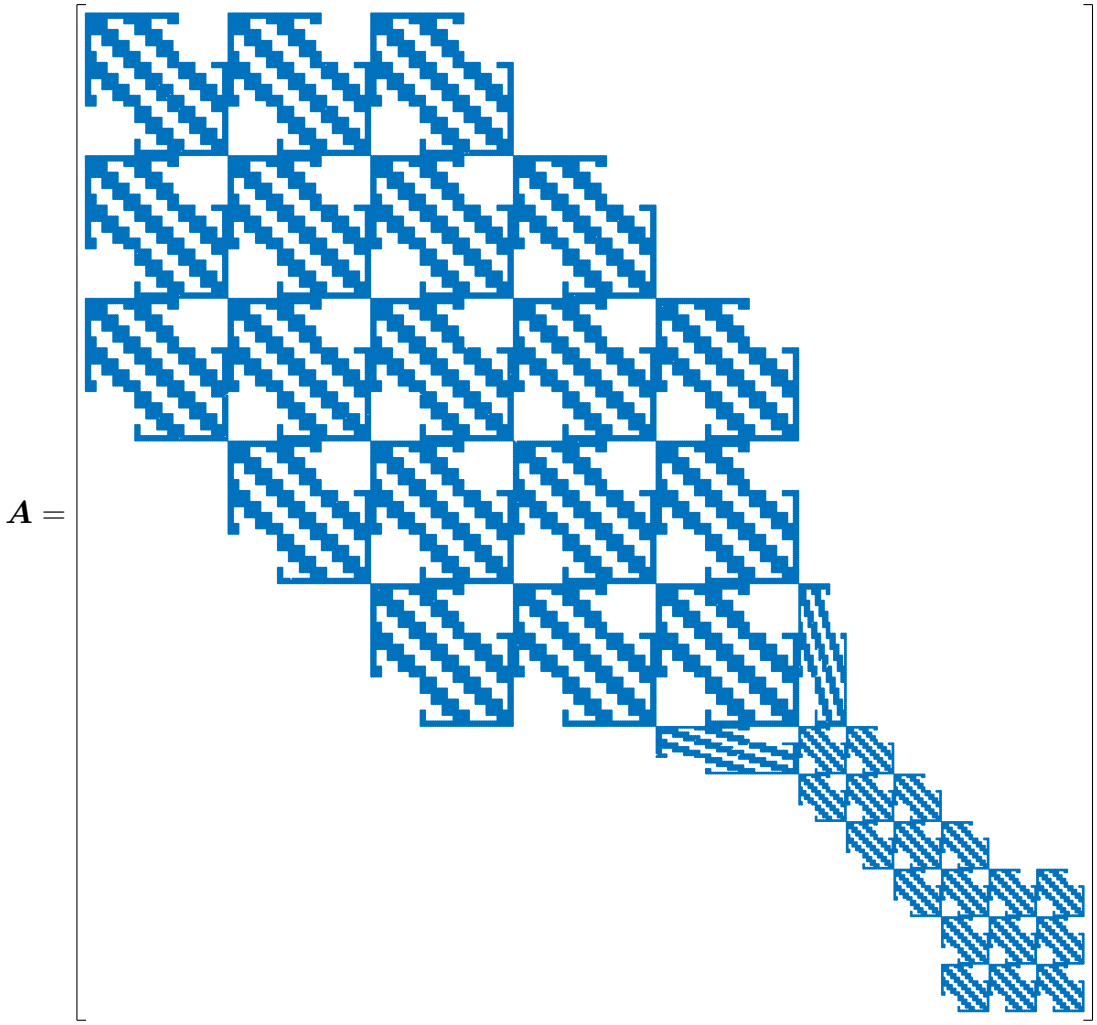


Figure 8.3: Elastic scattering on spherical shell: The same global matrix as in Figure 8.2, but with permutations in the solid part of the matrix to reduce bandwidth.

note that we get $d \cdot 1 + 1 \cdot d = 6$ such "squares" in the coupling area (recall that $d = 3$ is the number of physical dimensions we use). Without this coupling part, the system becomes decoupled, and one end up solving two separate problems in one go (this is actually a good test for developing the procedure of fluid-structure interaction problems).

In Figure 8.3 we show another configuration of the indexing of the NURBS basis function in the solid. Here the first three indices corresponds to the x, y and z component of the displacement field for the first NURBS basis function. This results in a smaller bandwidth. However, MATLAB uses its own permutation algorithm when a direct solver is applied, so the ordering is not so important.

8.6 Computation of backscattered pressure

Again we loop through the surface elements at Γ and use Equation (6.16) to compute the backscattered pressure. The main difference from the computation of the coupling conditions, is that we here need to compute the derivative. Thus, the full 3D NURBS is

needed in order to compute the Jacobian matrix.

If there are many angles to be analyzed, the two loops over these angles should be parallelized. However, if one only consider few angles (for instance the bistatic case when only one pair of α_s and β_s is considered), one should take the parallelization over the loop which runs through the surface elements.

Chapter 9

Numerical examples

We shall here present results obtained for the scattering on BeTSSi model 3 using hard walled boundary conditions (HWBC), and then continue with the fluid structure interaction (FSI) problem on the spherical shell. Unless stated otherwise, the results uses second order NURBS in all parameter directions, we set $N = 3$ (number of radial shape functions in the infinite elements) and we shall use the layers $r_m = mr_\mathrm{a}$ for $m = 1, \ldots, N$ (this is what Burnett suggested in [10]). No thorough analysis was made on the position of these layers, but when considering BeTSSi model 3, changes in r_m did not notably change the results (this was also commented by Burnett). Placing the layers very close to the object, or very far from the object did however reduce the accuracy of the results. So we shall use the convention after Burnett. Moreover, the position of the artificial boundary Γ_a is also lacking a thorough analysis. Some investigation revealed that a somewhat increase in the accuracy was obtained if the density of the elements remained constant. This however, drastically increased the computational time, and was thus not considered to be advantageous.

9.1 Scattering on BeTSSi model 3

We shall present both bistatic and monostatic results of scattering on BeTSSi model 3. The main theme in the results will be the pollution effect as the results become less accurate for high frequencies using the same mesh. It will illustrate the importance of mesh refinement for the increase in the frequency.

In all results, we have set the elevation angle of the incident wave and the far field point to be zero, that is $\beta_\mathrm{s} = 0$ and $\beta_\mathrm{f} = 0$.

Most of the results will be compared with results obtained from Ilkka Karasalo and Martin Østberg at FOI[1]. They have proposed several methods for solving the scattering problems including the boundary element method (FOI4). For BeTSSi model 3, they exploit the axisymmetry to reduce the computational time, such that the obtained results may not be compared to the obtained results in this thesis when conserning computational time. However, we shall use the results from FOI as reference solutions.

[1] FOI, Totalförsvarets forskningsinstitut, is the Swedish Defence Research Agency.

9.1.1 The near field

For illustrative purposes we start by plotting (the real part of) the near field for $f = 500$Hz and $f = 1$kHz, and for the aspect angles $\alpha_s = 240°$ and $\alpha_s = 300°$. In Figure 9.1 and Figure 9.2 we consider $f = 500$Hz with the aspect angles $\alpha_s = 240°$ and $\alpha_s = 300°$, respectively. The incident wave at aspect angle $\alpha_s = 240°$ arrives normal to the surface of BeTSSi model 3 only at one point, namely a point on the larger hemisphere. From here the scattered waves form a locally spherical wave pattern, as expected. What is peculiar about these plots is the "shadow" area behind BeTSSi model 3 where one should expect lower pressure. Instead we see waves forming with higher amplitude compared to the other scattered wave. This is because these waves do not represent the total pressure field[2], rather they represents the canceling part of the incident wave such that the total field indeed is reduced in this area. We call this area the shadow area, as the model casts a shadow behind itself. All of these effects are also observed when $\alpha_s = 300°$ in Figure 9.2. The corresponding two plots for $f = 1$kHz in Figure 9.3 and Figure 9.4 shows the same behavior, with a doubling in the number of waves. However, for higher frequencies, special lines of diffraction patterns emerges in the shadow area.

9.1.2 Bistatic scattering

Bistatic scattering consist of analyzing a single incident wave on the obstacle, and measure the scattered pressure in a series of far-field points. We shall consider bistatic scattering for incident waves with aspect angles $\alpha_s = 240°$ and $\alpha_s = 300°$, and for the frequencies 100Hz, 500Hz and 1kHz. For all these 6 cases we compare with the results obtained by FOI.

In Figure 9.5 and Figure 9.6 we consider the two cases for $f = 100$Hz. At such a low frequency, we observe very good results. Even mesh 1 yields a tolerable result. However, increasing the frequency to 500Hz we see in Figure 9.5 and Figure 9.6 that a corresponding increase in the number of elements is needed. In the final increase of the frequency to 1kHz the accuracy of the results fall correspondingly. This seems to be dominant in the area $\alpha_f \in [120°, 300°]$. The angle α_s lies in this interval, and as we shall see, the monostatic analysis on the same frequency reveals the same loss of accuracy for the full set of angles α_s. In other words, using the chosen set of parameters (mesh, polynomial order, etc.), considering the same aspect angle for the incident wave and the far-field point yields slower convergence. Note that mesh 5 is the most refined mesh available for a direct solver with the equipment at hand.

Finally, for illustration purposes, we plot the bistatic pressure also for a set of angles β_s which then results in a 3D plot. In Figure 9.11 we consider $\alpha_s = 240°$ and add the geometry of BeTSSi model 3 as a location reference (this geometry is scaled up in order to make it visible). In Figure 9.12 we plot the full set of elevation angles β_f. The similar case for $\alpha_s = 300°$ are depicted in Figure 9.13 and Figure 9.14.

[2] Which we here define to be the superposition of the incident field and the scattered pressure.

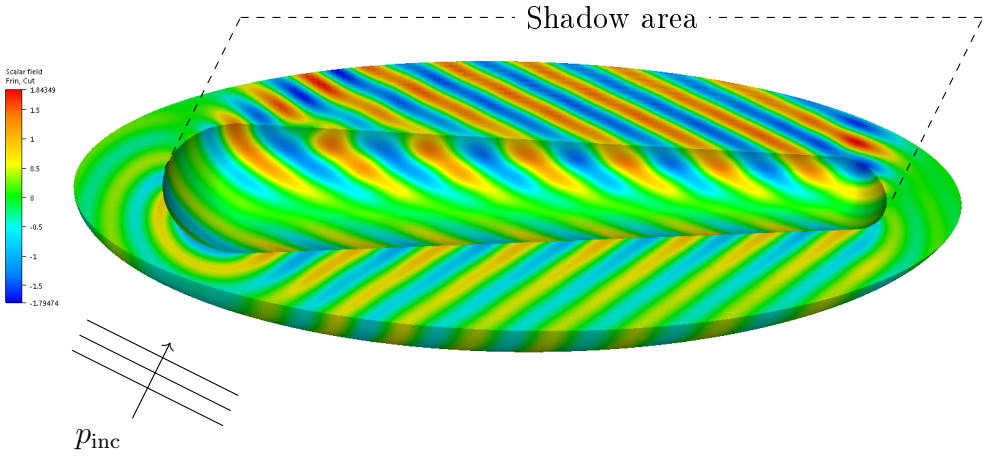


Figure 9.1: Scattering on BeTSSi model 3: The near field of BeTSSi model 3 with $f = 500\text{Hz}$ on mesh 5. The aspect angle for the incident wave is $\alpha_s = 240°$. Mesh 5 is illustrated in Figure 6.9.

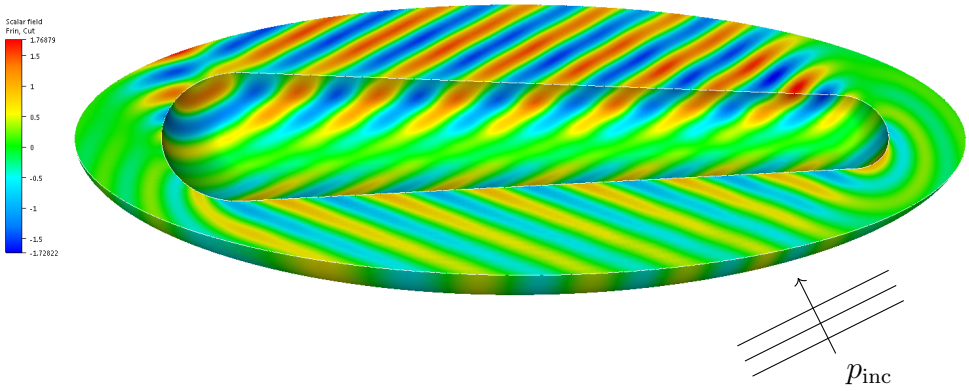


Figure 9.2: Scattering on BeTSSi model 3: The near field of BeTSSi model 3 with $f = 500\text{Hz}$ on mesh 5. The aspect angle for the incident wave is $\alpha_s = 300°$. Mesh 5 is illustrated in Figure 6.9.

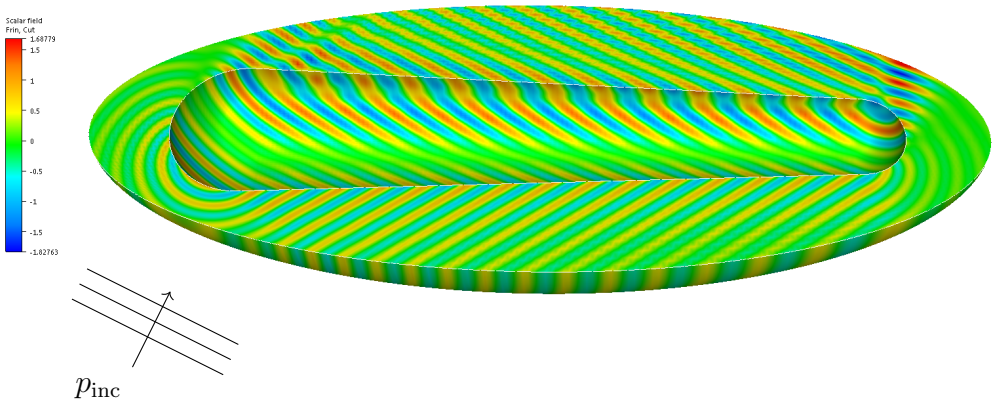


Figure 9.3: Scattering on BeTSSi model 3: The near field of BeTSSi model 3 with $f = 1\text{kHz}$ on mesh 5. The aspect angle for the incident wave is $\alpha_s = 240°$. Mesh 5 is illustrated in Figure 6.9.

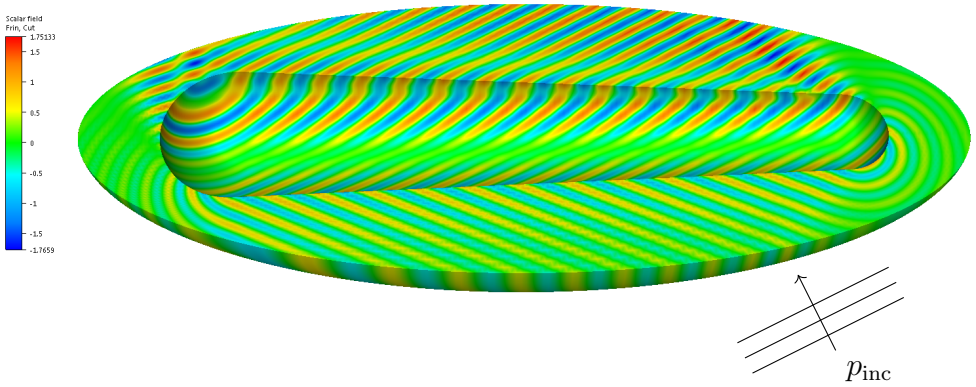


Figure 9.4: Scattering on BeTSSi model 3: The near field of BeTSSi model 3 with $f = 1\text{kHz}$ on mesh 5. The aspect angle for the incident wave is $\alpha_s = 300°$. Mesh 5 is illustrated in Figure 6.9.

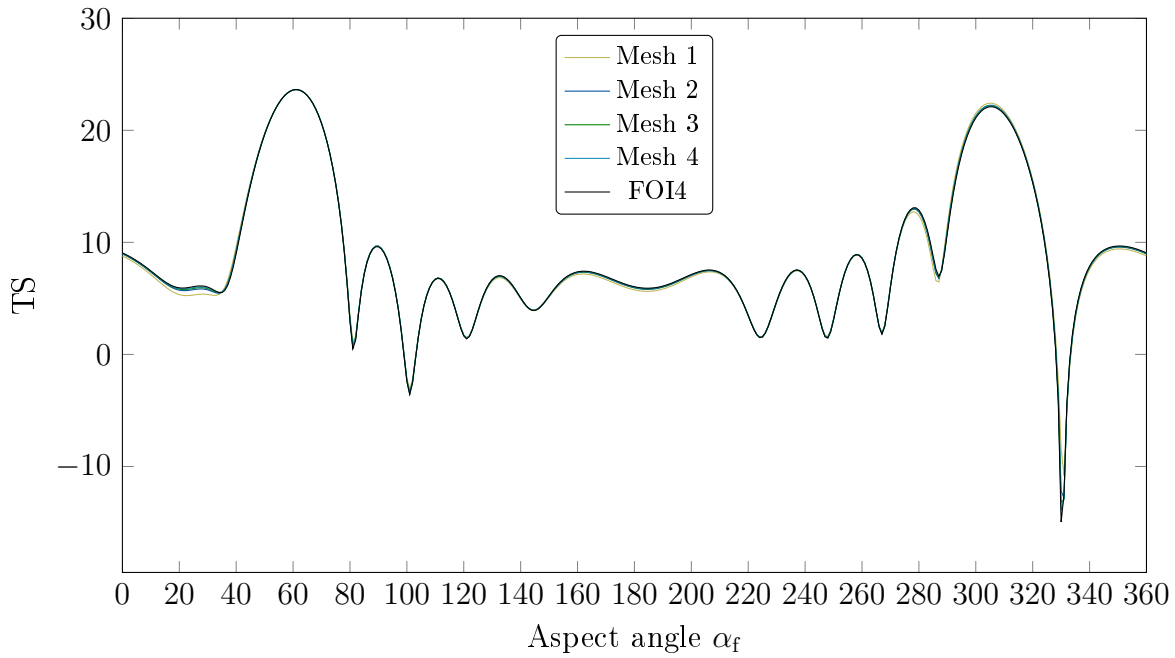


Figure 9.5: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100$Hz. The aspect angle for the incident wave is $\alpha_s = 240°$. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

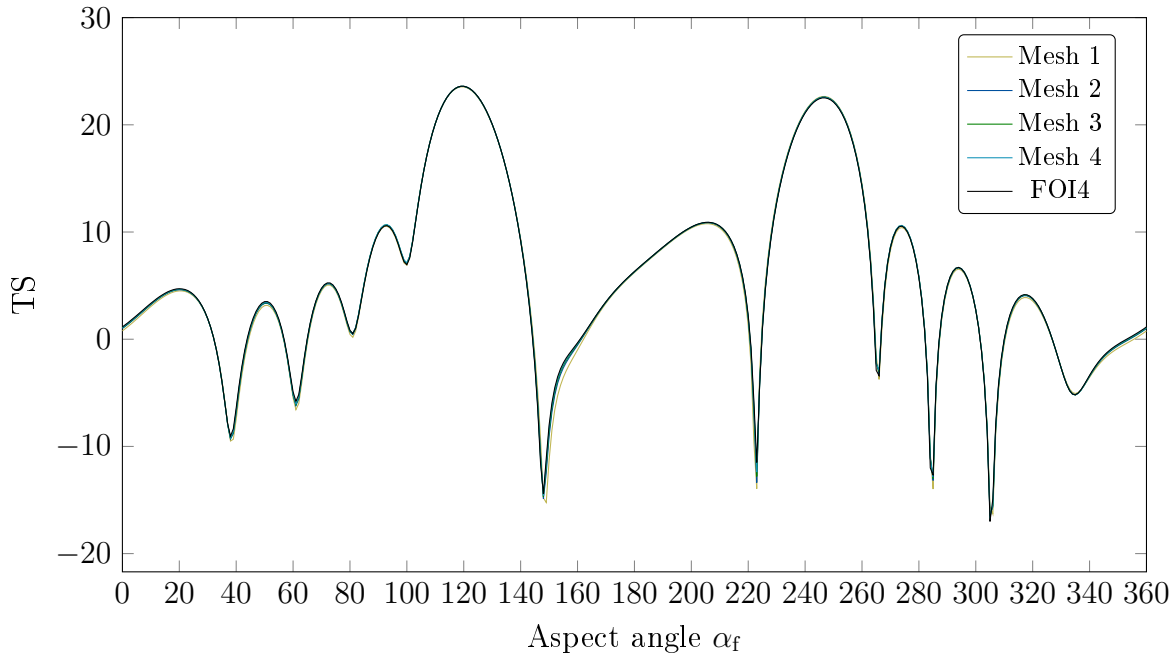


Figure 9.6: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100$Hz. The aspect angle for the incident wave is $\alpha_s = 300°$. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

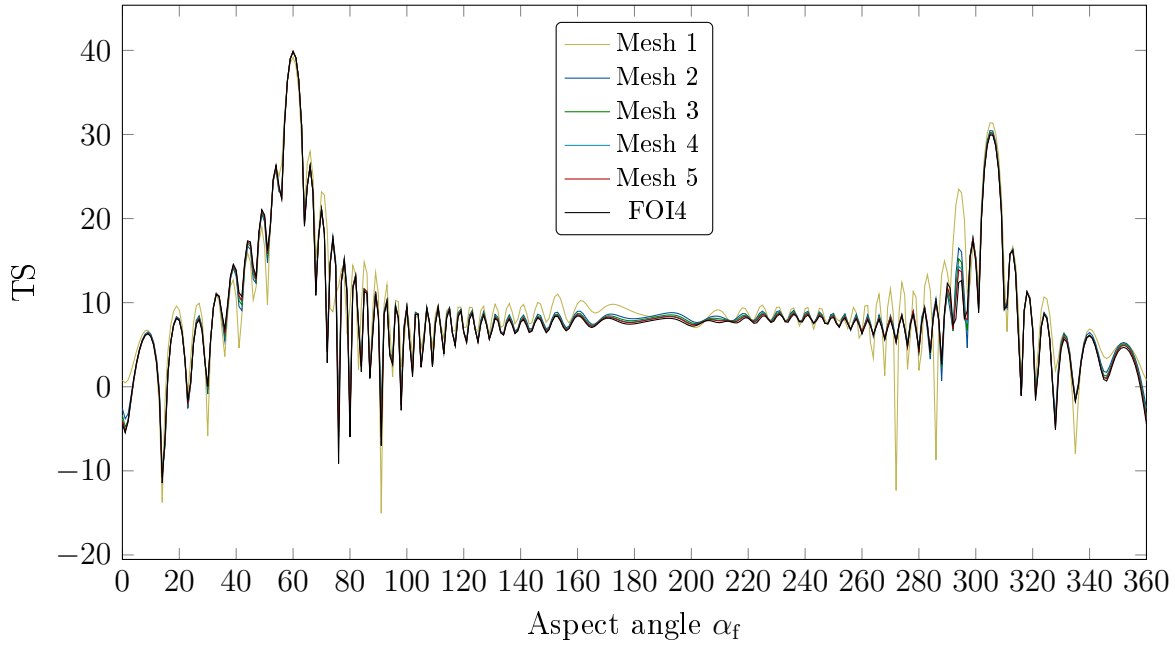


Figure 9.7: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 500$Hz. The aspect angle for the incident wave is $\alpha_s = 240°$. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

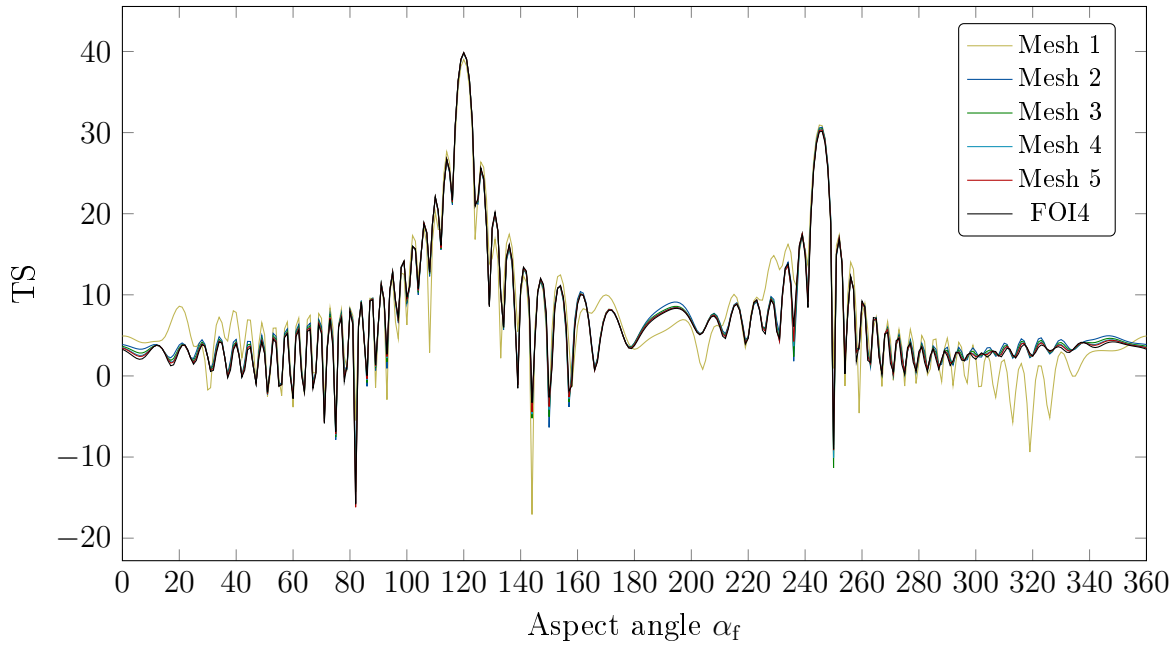


Figure 9.8: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 500$Hz. The aspect angle for the incident wave is $\alpha_s = 300°$. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

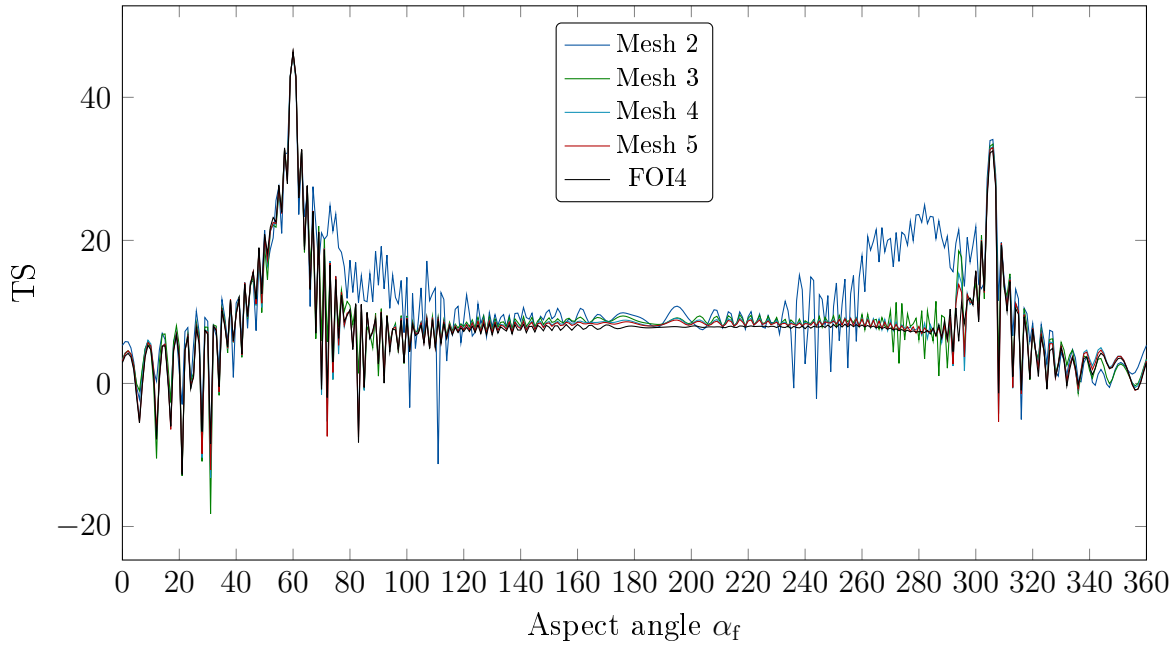


Figure 9.9: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 1$kHz. The aspect angle for the incident wave is $\alpha_s = 240°$. Mesh 2, 3, 4 and 5 are illustrated in Figure 6.9.

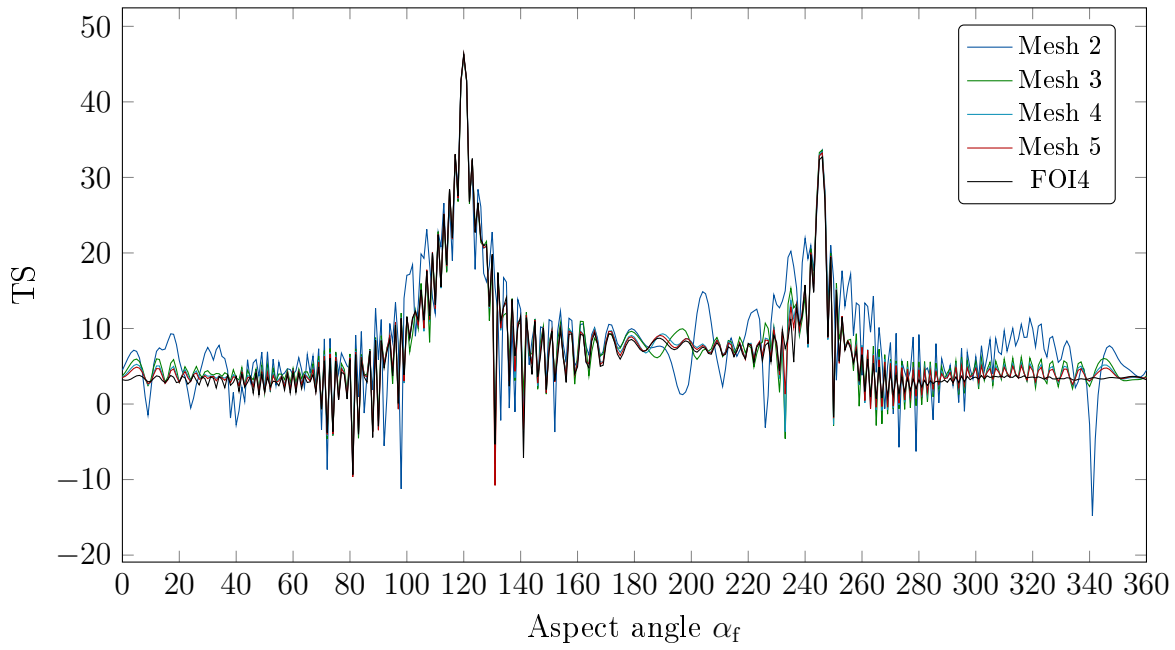


Figure 9.10: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 1$kHz. The aspect angle for the incident wave is $\alpha_s = 300°$. Mesh 2, 3, 4 and 5 are illustrated in Figure 6.9.

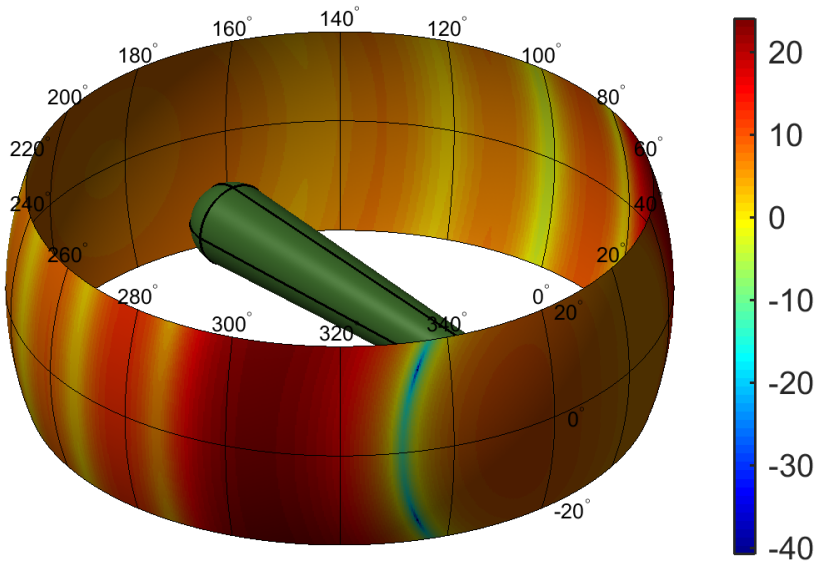


Figure 9.11: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100\text{Hz}$ on mesh 1. The aspect angle for the incident wave is $\alpha_s = 240°$. The plot spans elevation angles β_f from $-20°$ to $20°$ and the whole range of aspect angles in steps of $0.25°$ (results in $232,001$ evaluation points). Mesh 1 is illustrated in Figure 6.7c.

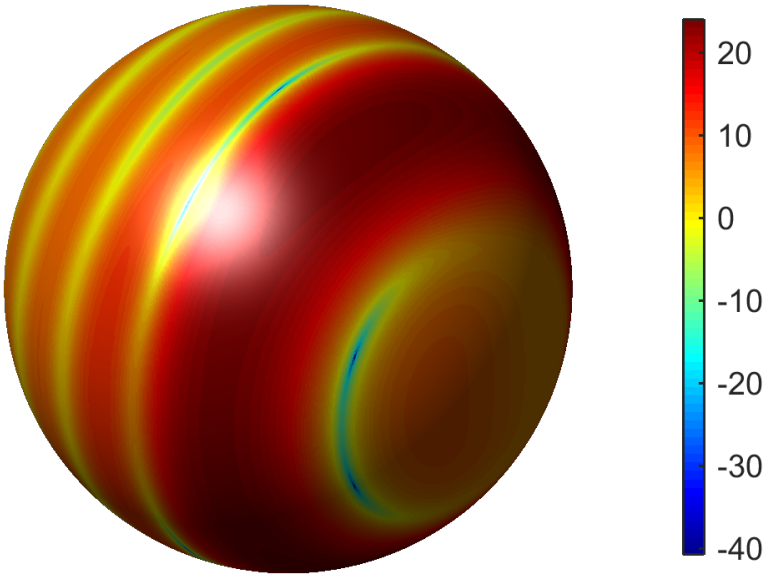


Figure 9.12: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100\text{Hz}$ on mesh 1. The aspect angle for the incident wave is $\alpha_s = 240°$. The plot spans the whole range of elevation angles and of aspect angles in steps of $0.25°$ (results in $1,038,961$ evaluation points). Mesh 1 is illustrated in Figure 6.7c.

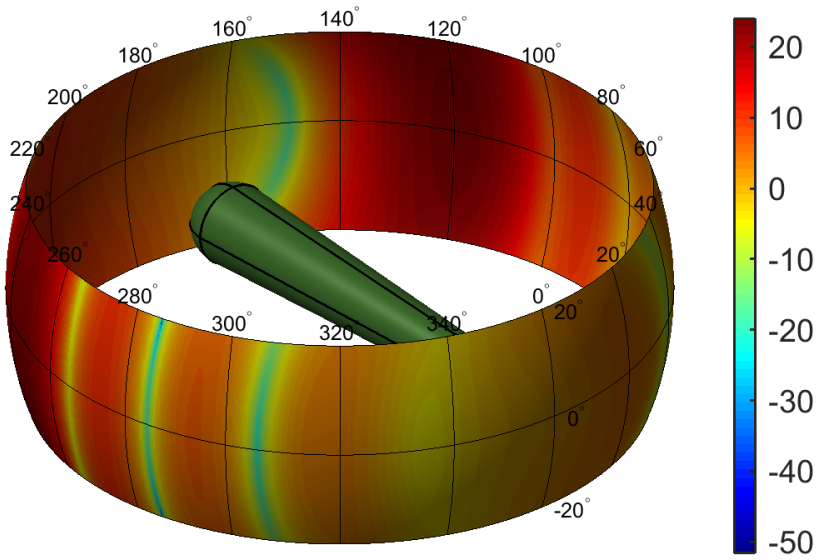


Figure 9.13: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100\text{Hz}$ on mesh 1. The aspect angle for the incident wave is $\alpha_s = 300°$. The plot spans elevation angles β_f from $-20°$ to $20°$ and the whole range of aspect angles in steps of $0.25°$ (results in $232,001$ evaluation points). Mesh 1 is illustrated in Figure 6.7c.

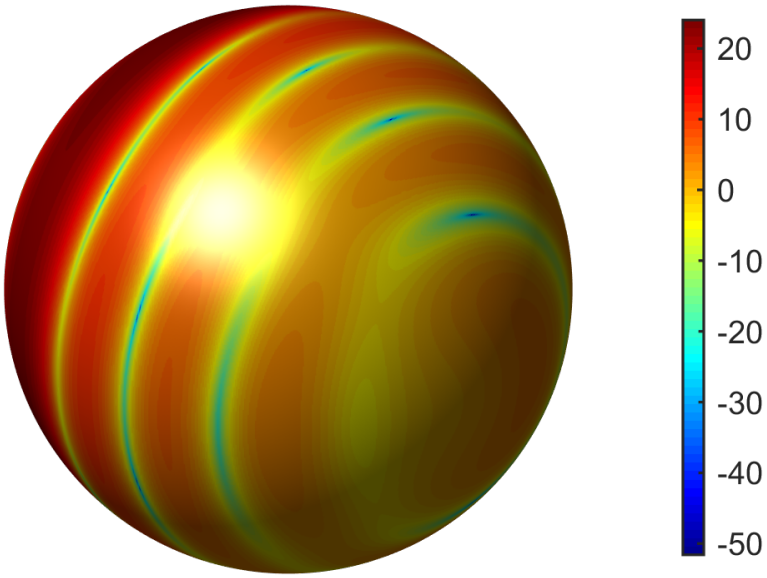


Figure 9.14: Bistatic scattering on BeTSSi model 3: The bistatic TS of BeTSSi model 3 with $f = 100\text{Hz}$ on mesh 1. The aspect angle for the incident wave is $\alpha_s = 300°$. The plot spans the whole range of elevation angles and of aspect angles in steps of $0.25°$ (results in $1,038,961$ evaluation points). Mesh 1 is illustrated in Figure 6.7c.

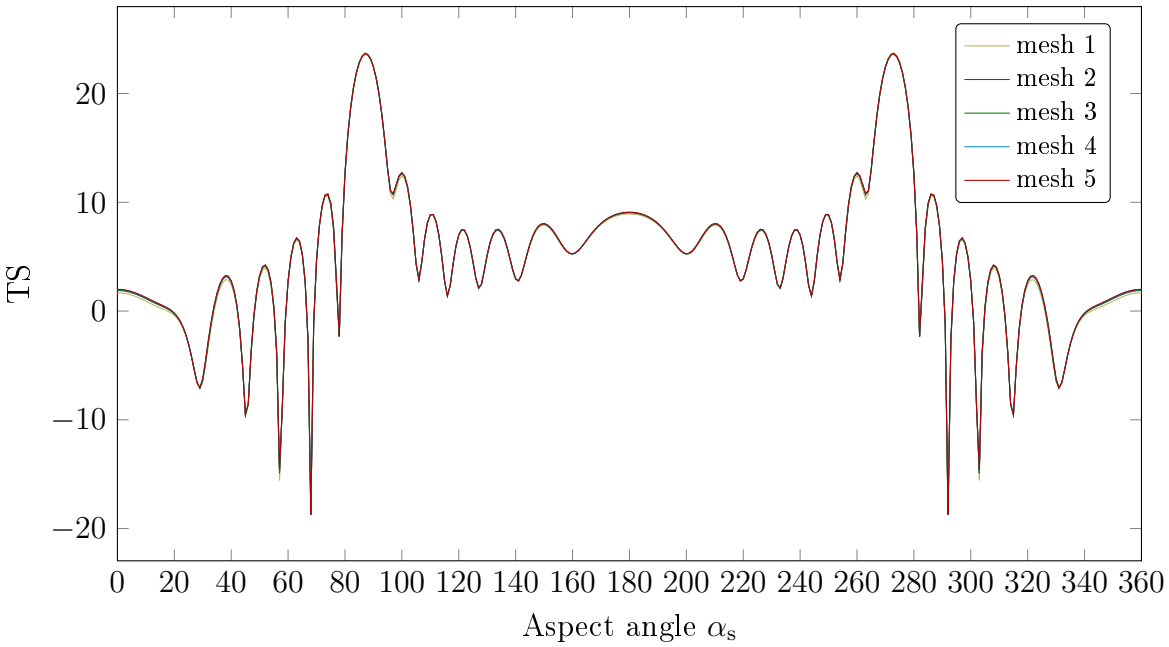


Figure 9.15: Monostatic scattering on BeTSSi model 3: The monostatic TS of BeTSSi model 3 with $f = 100$Hz. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

9.1.3 Monostatic scattering

In monostatic scattering we consider the case $\alpha_f = \alpha_s$ and $\beta_f = \beta_s$. Due to the geometry of BeTSSi model 3, one expects peaks at incident waves which hits the scatterer normal to the broadsides (as noted in [12]). A simple trigonometric calculation then reveals these angles to be

$$\alpha_{s,max}^{(1)} = \frac{\pi}{2} - \tan^{-1}\left(\frac{R_{o1} - R_{o2}}{L - R_{o1} - R_{o2}}\right) = 1.522054475 = 87.20729763°,$$
$$\alpha_{s,max}^{(2)} = \frac{3\pi}{2} + \tan^{-1}\left(\frac{R_{o1} - R_{o2}}{L - R_{o1} - R_{o2}}\right) = 4.761130832 = 272.7927024°,$$

where $R_{o1} = 5$ and $R_{o2} = 3$ is the outer radius of the larger and smaller hemisphere, respectively, and $L = 49$ is the total length of BeTSSi model 3 (including the hemispheres). Both of these predictions are verified in the results.

In Figure 9.15 we consider monostatic scattering with $f = 100$Hz. For such a low frequencies we obtain very good results, even for mesh 1. Increasing the frequency to $f = 500$Hz, we observe in Figure 9.16 that we need a corresponding increase in the number of elements to reach the same accuracy. Unfortunately we have no data sets from FOI to compare with for these two cases. Finally, increasing the frequency to 1kHz we can compare the results to FOI (Figure 9.17), and we do not reach convergence on mesh 5. It is assumed that the resolution of the mesh is too low to obtain good results at this frequency. That is, our results suffers from the pollution effect.

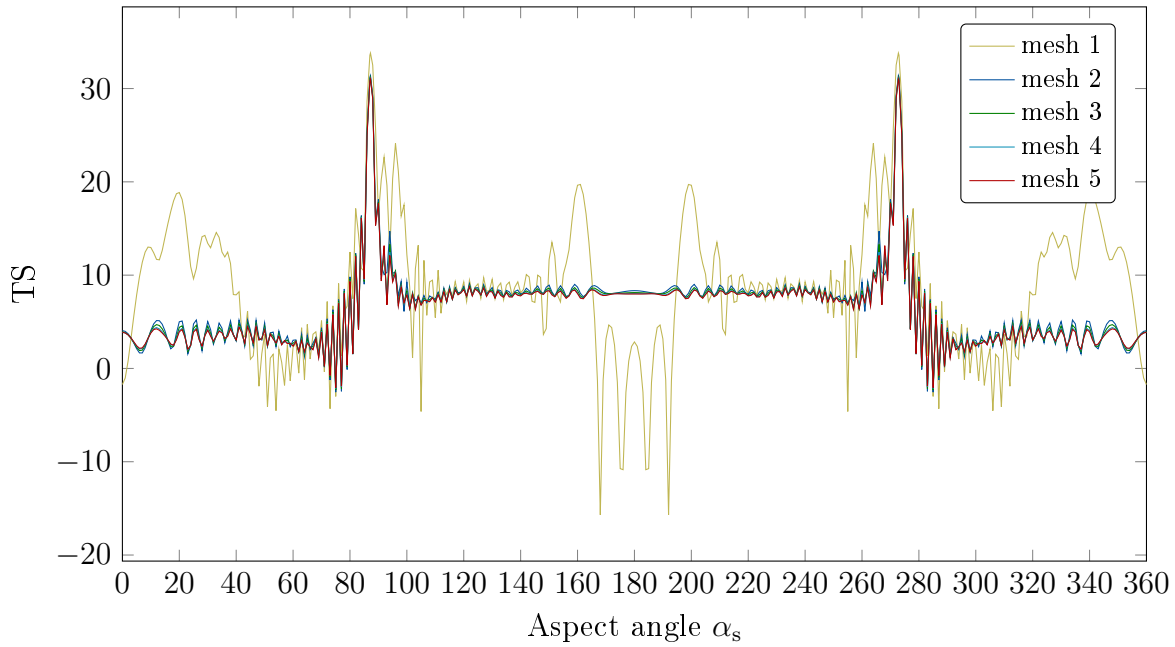


Figure 9.16: Monostatic scattering on BeTSSi model 3: The monostatic TS of BeTSSi model 3 with $f = 500$Hz. Mesh 1 and mesh 2, 3, 4 and 5 are illustrated in Figure 6.7c and Figure 6.9, respectively.

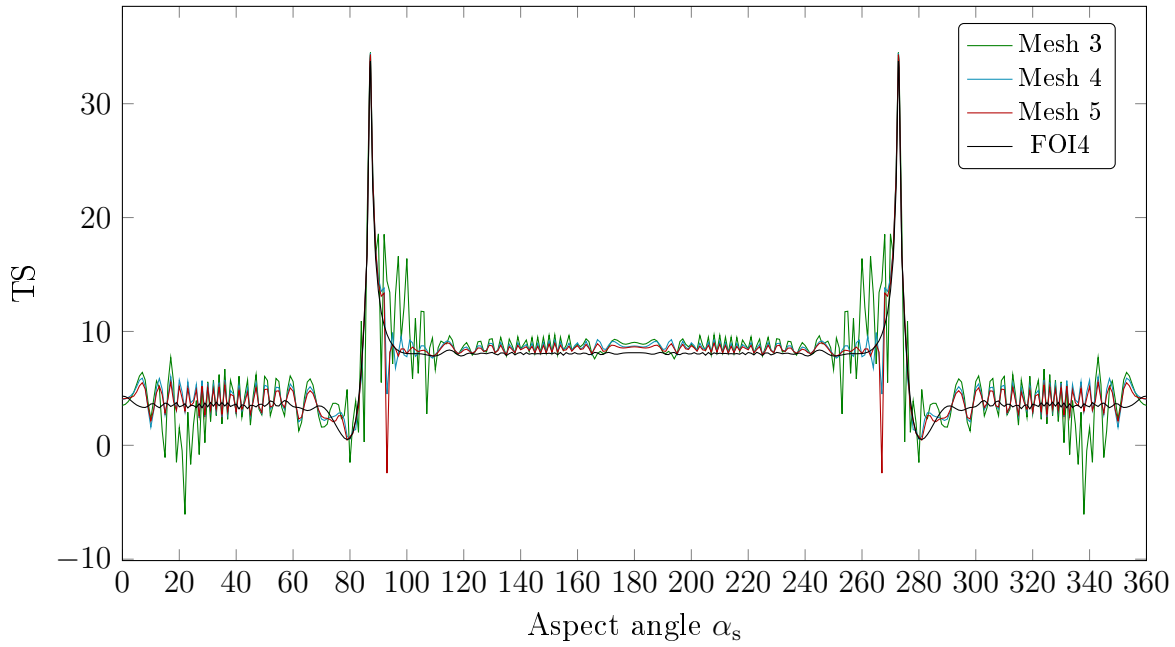


Figure 9.17: Monostatic scattering on BeTSSi model 3: The monostatic TS of BeTSSi model 3 with $f = 1$kHz. Mesh 3, 4 and 5 are illustrated in Figure 6.9.

9.2 FSI of the spherical shell

We now turn to the fluid-structure interaction (FSI) problem on a spherical shell where we want to compare FEA and IGA. It turns out that the results on the spherical shell using FEA from [3] has been lost[3], so it is not so easy to visualize the comparison between the two methods. Ihlenburg exploits the symmetry in the problem by eliminating one dimension (namely the azimuth dimension) with the use of a Fourier expansion. This has not been done in this thesis as the aim is to develop a generic program which could handle non-symmetric geometries. Thus, any comparison with the data sets from Ihlenburg would not be perfectly valid.

In order to compare the finite element method (FEM) and IGA on the scattering problem on the spherical shell, we shall transform the NURBS mesh to a classical FEM mesh. We start with an arbitrary IGA mesh, for instance mesh 3 and 4 in Figure 9.18a and Figure 9.18c. These two meshes will be transformed to the meshes depicted in Figure 9.18b and Figure 9.18d, respectively. This is done by repeating every knot in the IGA mesh such that every internal knot has multiplicity p in the ξ-direction and correspondingly in the other two directions. To get the polygon elements, we simply project every control point which does not lie on the surface, to the surface of its corresponding polygon element. Finally, all weights are set to be 1. Due to the repeating of knots in the FEM mesh, we increase the number of degrees of freedom (dofs). We want to do analysis of FEM and IGA where both methods use the same number of dofs. We thus construct matching IGA and FEM meshes w.r.t. dofs (see Figure 9.19). We shall call these IGA meshes mesh 3' and mesh 4'.

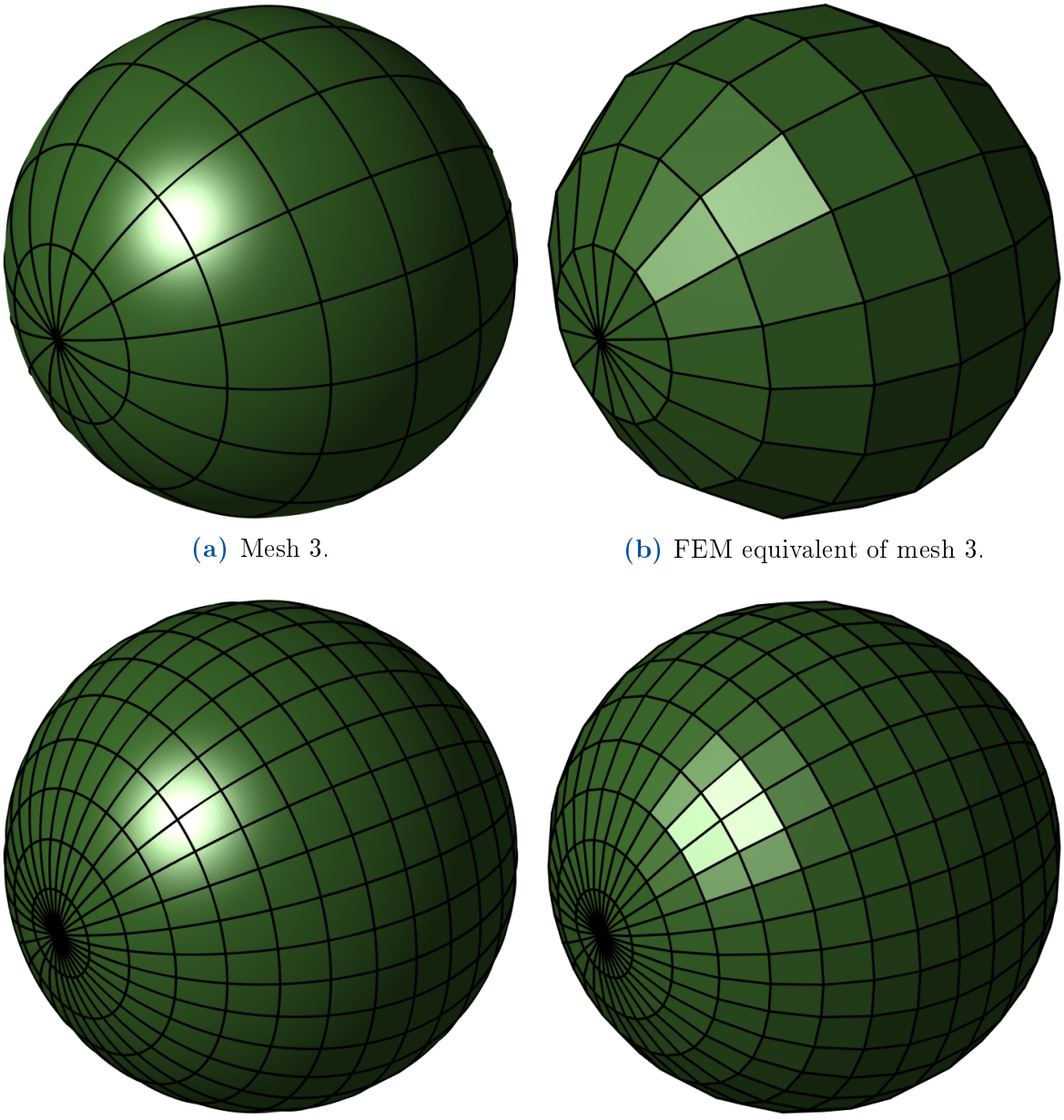
It should be noted that the FEM analysis will be completed with the Bernstein basis instead of the classical Lagrange basis. However, both of these set of functions spans the same spaces, such that the results should be identical.

Unless stated otherwise, we place the artificial boundary at $r_a = 6$ and use $N = 3$ radial shape functions in the infinite elements. Moreover, we shall mainly use the BGU formulation for the infinite elements.

In Figure 9.20 and Figure 9.21 we compare FEA and IGA on mesh 3/4 and mesh 3'/4', respectively. In addition we add the results from the IGA mesh 3 with repeated knots, to investigate the importance of the exact geometry. We shall refer to this as the IGA mesh with C^0-continuity.

At the coarse FEM mesh 3 and IGA mesh 3' we first observe a large deviation from the exact solution at higher wave numbers for FEA, but also for the IGA C^0-continuity mesh. The corresponding IGA C^1-continuity mesh has much better results in this area. We continue to observe a shift in the lower eigenmodes for FEA. This shift is also present for the IGA mesh with C^0-continuity when comparing to IGA mesh 3 (all though not to the same extent). We could therefore argue that reduction in the continuity and the non-exactness of the geometry, shifts the eigenmodes to the right. One might wonder whether or not this is due to the same effect observed from the vibration of the one dimensional elastic rod in Section 5.1. It should be noted the results for the IGA mesh with C^1-continuity are also shifted to the right compared with the exact solution, but the shift is reduced. This could be due to the fact that we have C^0-continuity at $\eta = 0.5$

[3] Private communication with Frank Ihlenburg, May 2015



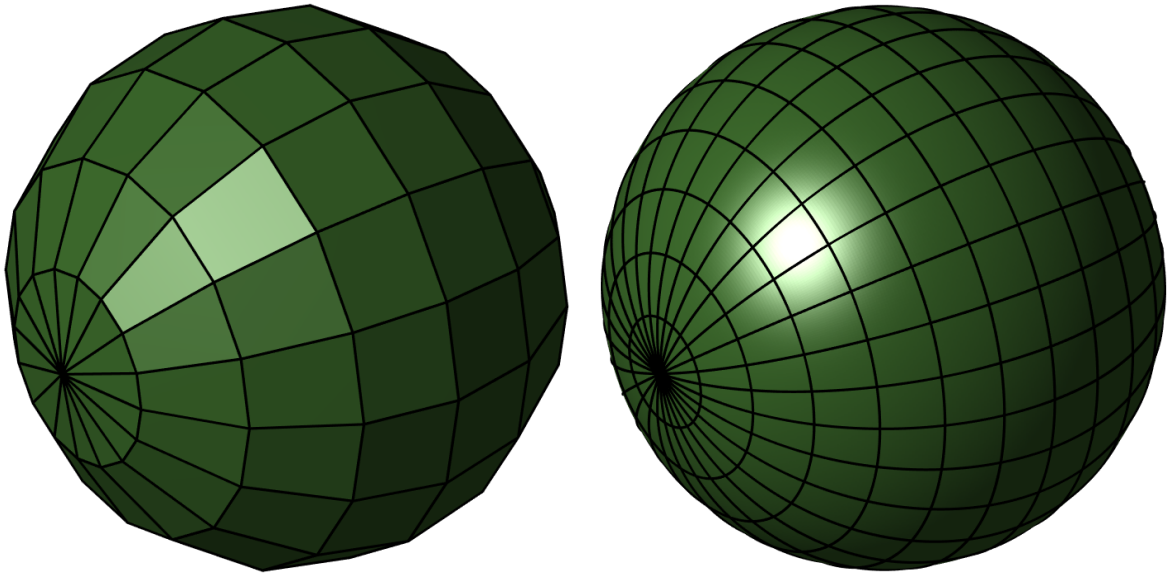
(a) Mesh 3.

(b) FEM equivalent of mesh 3.

(c) Mesh 4.

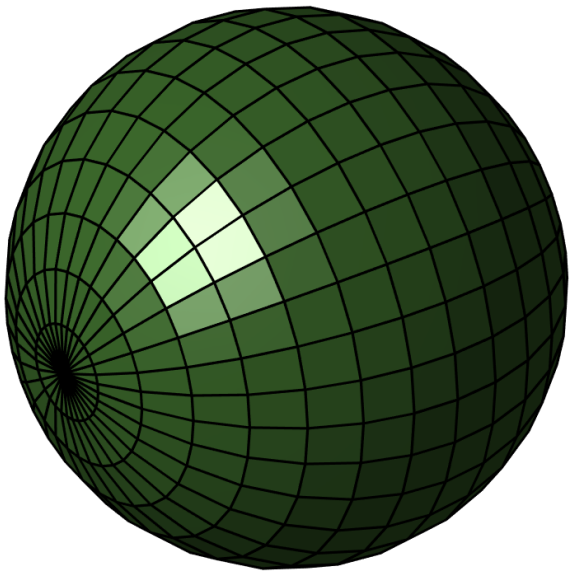
(d) FEM equivalent of mesh 4.

Figure 9.18: FSI of the spherical shell: Mesh 3 and 4 for spherical shell with its corresponding equivalent FEM mesh.

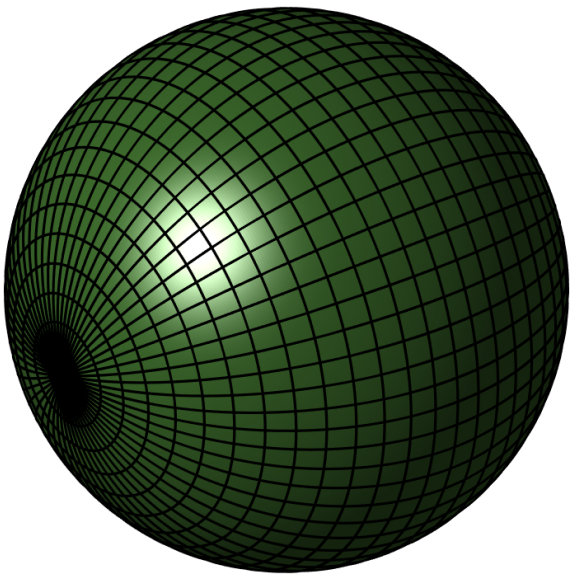


(a) FEM mesh 3.

(b) IGA mesh 3'.



(c) FEM mesh 4.



(d) IGA mesh 4'.

Figure 9.19: FSI of the spherical shell: FEM mesh 3 and 4 with corresponding IGA mesh with matching number of dofs. For each knot that has been inserted in the ξ direction in the FEM mesh (and repeated p), it must be inserted p knots in the original mesh to match the number of dofs (correspondingly in the other directions).

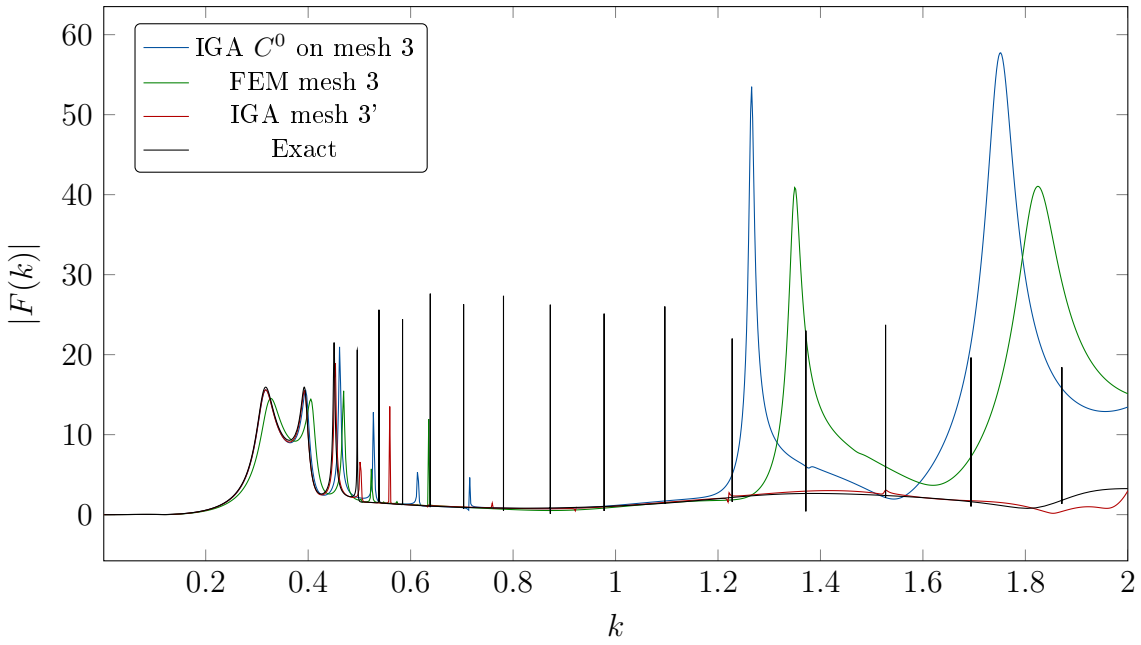


Figure 9.20: FSI of the spherical shell: A comparison of FEM and IGA on mesh 3 and mesh 3', respectively. IGA C^0 on mesh 3, FEM mesh 3 and IGA mesh 3' are depicted in Figure 9.18a, Figure 9.18b and Figure 9.19b, respectively. The exact solution is given in Equation (3.2).

also for these meshes, but this is because the NURBS geometry is constructed in such a way[4].

In Figure 9.21 we observe better results from all three cases, but still we can see a shift in the eigenmodes when the continuity is reduced. In Figure 9.22 we zoom in on the first 4 eigenmodes where we can observe the shift to be greatest for FEA. The shift is also here observed for the C^0-continuity mesh, and the magnitude of the shift grows for higher eigenmodes. The shift is therefore not connected to the exactness of the geometry alone, it is also connected to the continuity of the basis functions.

We shall end the analysis on the spherical shell by considering convergence. We shall do this by constructing a pivot mesh which all results will be compared to. From the coarsest mesh for the spherical shell (Figure 5.4a), we insert (uniformly) two knots in the ζ-direction for the fluid. No knots have been inserted in the ζ-direction for the solid. Due to axisymmetry, only 12 knots are inserted in the ξ-direction[5] (for both solid and fluid). Finally, 60 knots are inserted in η-direction for both the fluid and the solid.

In Figure 9.23 we order elevate[6] the NURBS order in ζ-direction (only) in the fluid. Not only is the accuracy greatly increased from order 1 to 2, but also from 2 to 3, indicating the importance of good resolution in the fluid. By comparison in Figure 9.23, an corresponding order elevation in the solid hardly improves the result, at least not from

[4] We also have C^0-continuity for $\xi = 0.25, 0.5, 0.75$, but the solution is constant in this direction, so it should not matter.

[5] The reason to insert any knots at all in this case, is to obtain more Gauss points for the integration of the constant functions in this direction. It could also have been solved by using more than $p+1$ Gauss points in the ξ-direction.

[6] Note that we order elevate before inserting any knots.

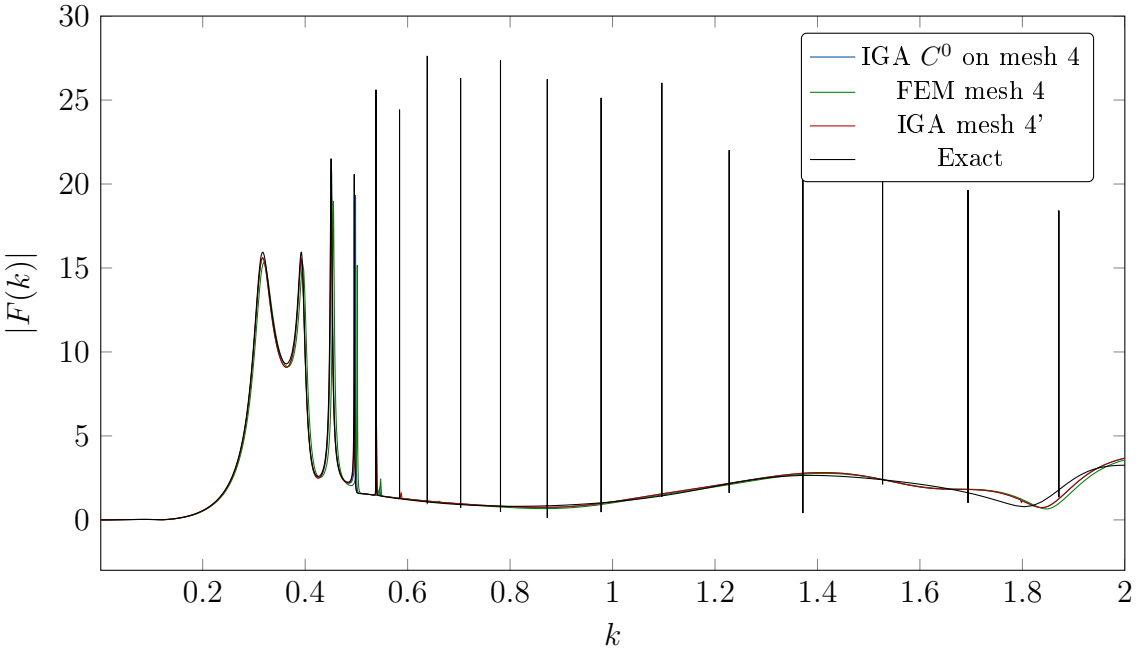


Figure 9.21: FSI of the spherical shell: A comparison of FEM and IGA on mesh 4 and mesh 4', respectively. IGA mesh IGA C^0 on 4, FEM mesh 4 and mesh 4' are depicted in Figure 9.18c, Figure 9.18d and Figure 9.19d, respectively. The exact solution is given in Equation (3.2).

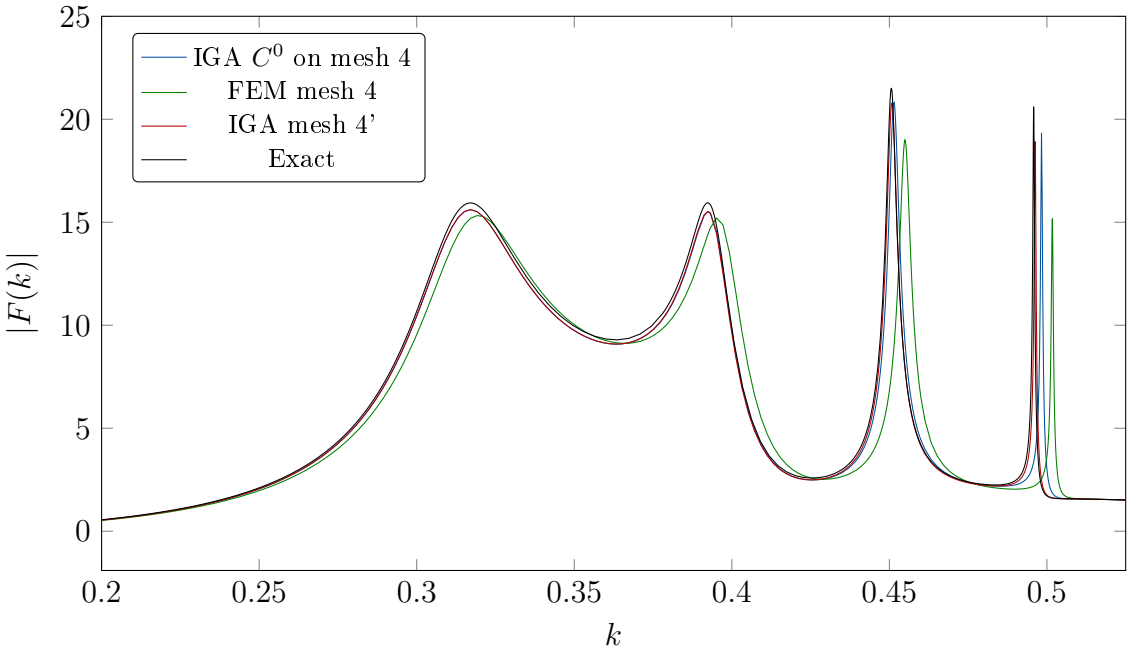


Figure 9.22: FSI of the spherical shell: Close-up on the comparison of FEM and IGA on mesh 4 and mesh 4', respectively. IGA mesh C^0 on 4, FEM mesh 4 and IGA mesh 4' are depicted in Figure 9.18c, Figure 9.18d and Figure 9.19d, respectively. The exact solution is given in Equation (3.2).

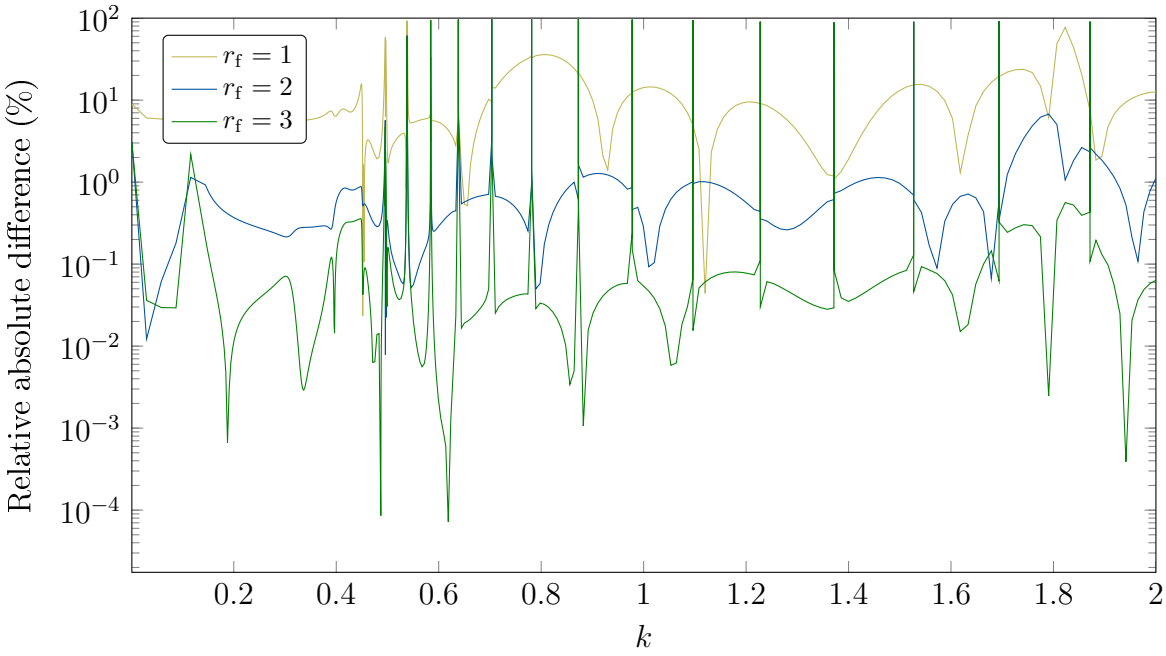


Figure 9.23: FSI of the spherical shell: Convergence plot considering order elevation in fluid (radial direction). The error is calculated by Equation (3.3) based on the exact solution in Equation (3.2).

the second order NURBS used in the pivot mesh.

Continuing the analysis on the infinite elements, we observe the importance of using more than one basis function in the infinite elements in Figure 9.25. No improvement was made by increasing this number to 6 using the other parameters from the pivot mesh. In Figure 9.26 we compare the different infinite elements formulations. We observe the unconjugated versions to be superior to the conjugated versions.

9.3 General remarks

In [10] Burnett suggest that the artificial boundary Γ_a should be placed (on average) $\frac{\lambda}{2}$ away from the object. That is, he suggest that this boundary to be placed closer at higher frequencies. This rule would be hard to impose on BeTSSi model 3 simply because BeTSSi model 3 differs to much from a prolate spheroid. This is a drawback, as it could be easy to draw a convex surface around BeTSSi model 3 which would nearly be a constant distance from the object surface Γ. The promising method of PML (Perfectly matched layers) after Bérenger (in [29] and [30]) manages to do just that, create a perfectly matched layer of fluid between Γ_a and Γ. In principle, this should be possible for the infinite element method (as long as Γ_a is made convex). However, this requires the use of a more general coordinate system (rather then the prolate spheroidal coordinate system). This topic has been somewhat investigated in Appendix B.1.1.

As we did not achieve completely satisfactory accuracy in the numerical simulation of the BeTSSi model 3 using HWBC, we did not make anyattempt to do the full FSI problem of this model. This problem consist of BeTSSi model 3 being filled with water, such that

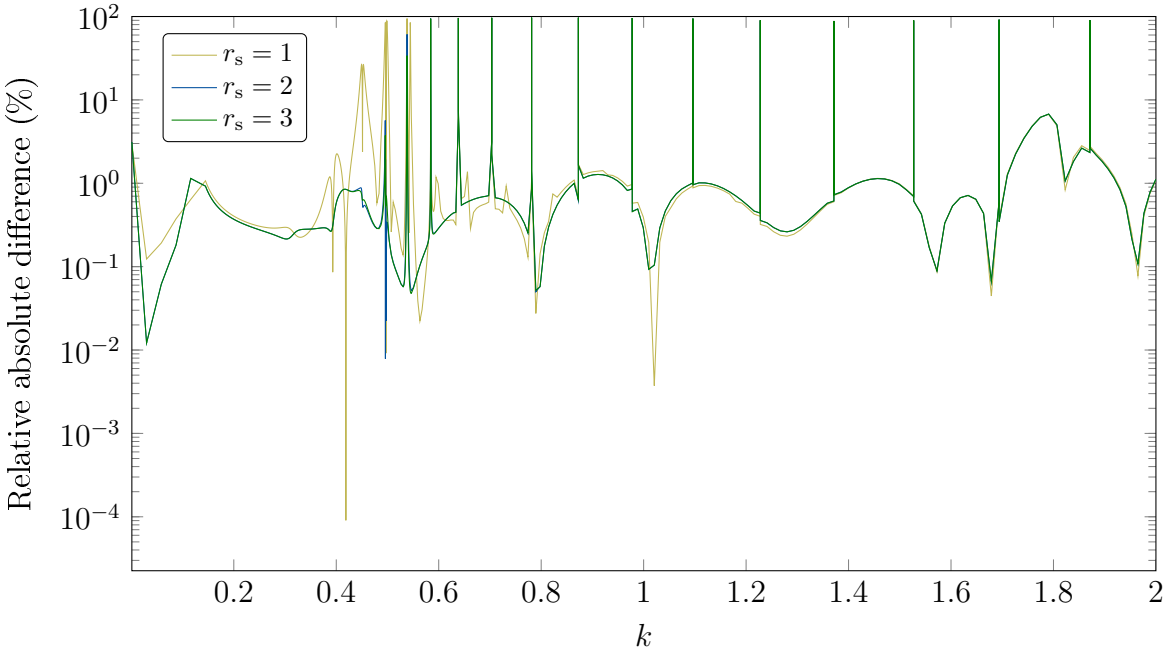


Figure 9.24: FSI of the spherical shell: Convergence plot considering order elevation in solid. The error is calculated by Equation (3.3) based on the exact solution in Equation (3.2).

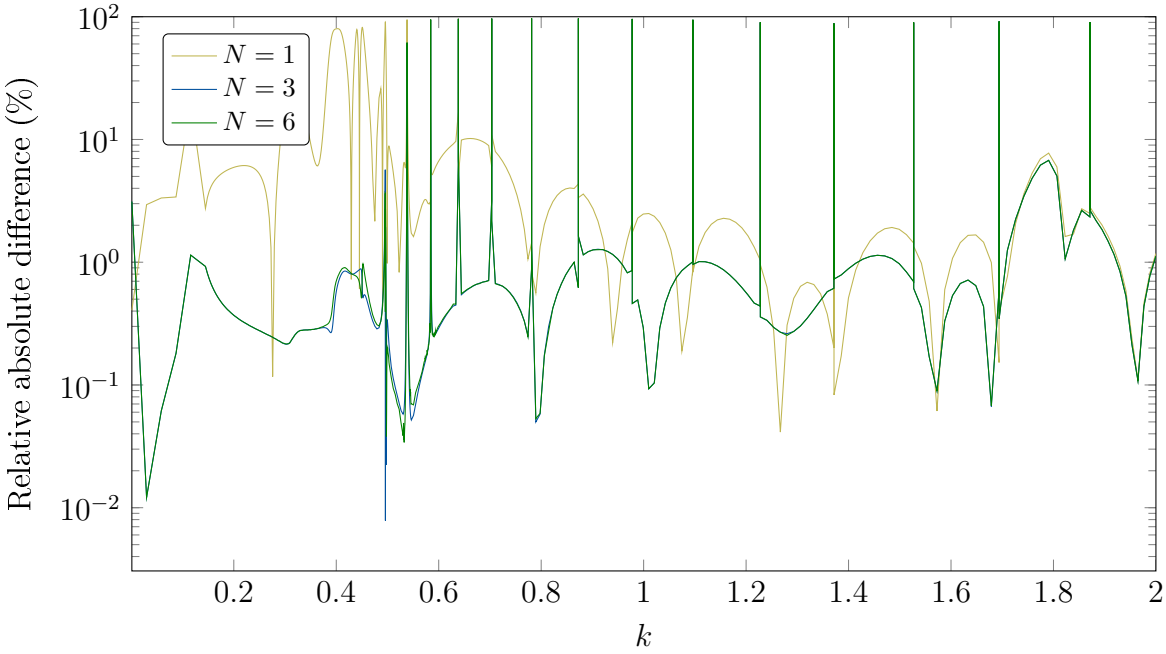


Figure 9.25: FSI of the spherical shell: Enrichment of basis functions in radial direction in the infinite elements. The error is calculated by Equation (3.3) based on the exact solution in Equation (3.2).

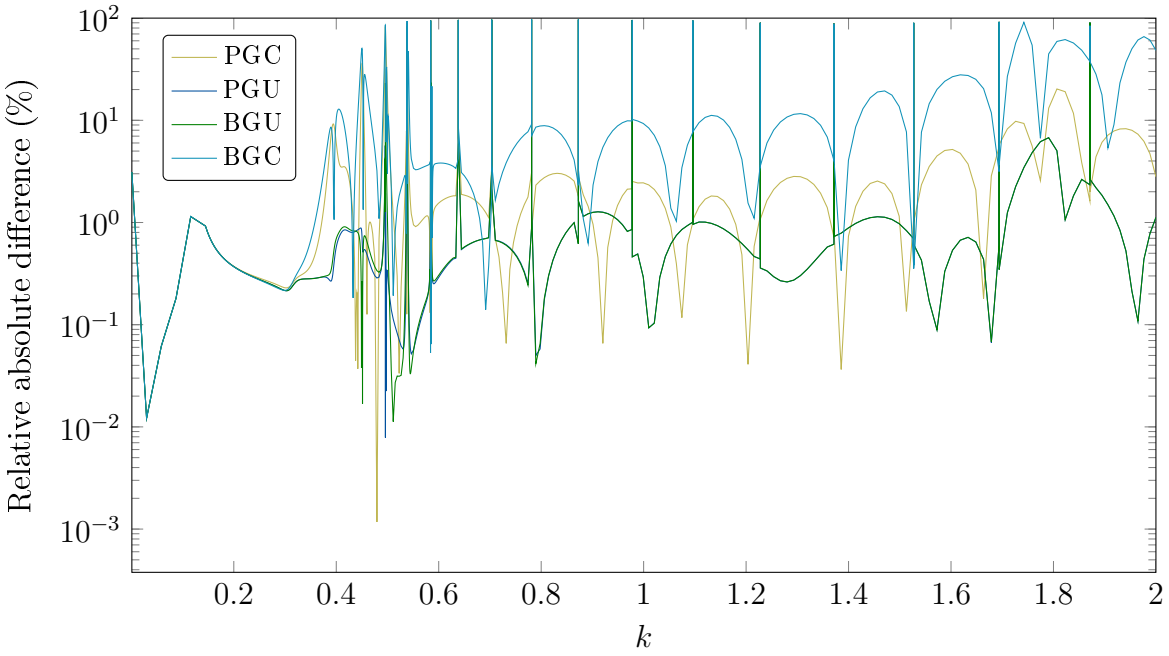


Figure 9.26: FSI of the spherical shell: Difference in the infinite element formulation. The error is calculated by Equation (3.3) based on the exact solution in Equation (3.2).

we have a interaction between fluid and the solid both inside and outside the shell. We must thus not only add structural elements, but also fluid elements inside the shell. The number of elements for mesh 5 (M3) then easily rises above one million, and will thus be outside the capacity of our direct solver.

The simulation using mesh 5 with HWBC has been using a server with 256GB available memory. With 700,000 elements (mesh 5), about 50% of this memory is used. The memory consumption is very expensive as the system of linear equation is solved by a direct solver. In MATLAB, a LU-factorization is made, and this is very memory consuming. An iterative iterative solver should be implemented to solve the issue. Not only will the memory consumption problem be eliminated, but we should also be able to speed up the solution process, as we do not require machine epsilon precision of the solution (after all, a 1% error in the result is considered to be very good). For classical elasticity problems on bounded domain, the global matrix is symmetric and positive definite (SPD). However, when introducing the infinite elements, the mass matrix is no longer positive definite, and neither is the resulting global matrix. This is the reason we may not use Cholesky factorization as a direct method. Moreover, this is the reason we we may not use the conjugate gradient (CG) method as this method also requires the matrix to be SPD. We must therefore go to other more general iteration techniques which typically have slower convergence rates. MATLAB has a built in routine for the GMRES method, which is designed for large sparse systems. Although CG cannot be used, there exist CG similar methods which does not require the matrix to be positive definite. Some of these methods are compared in [39], BiCGStab is also a natural choice to be investigated. The global matrix often turns out to be badly conditioned for iterative techniques. One should apply some preconditioning to this matrix before applying for instance GMRES. This typically results in a comprehensive analysis, which we shall suggest as future work.

Chapter 10

Conclusion

We have observed that IGA is well suited for finding eigenvalues of elastodynamical problems. Results from [2] were replicated, and corresponding results on the spherical shell turn out to result in the same type of convergence on a coarse mesh. Such convergence would be hard to obtain using the FEA on such a coarse mesh, as the exactness of the geometry then plays an important role.

We have seen that IGA and the four presented infinite element formulations work very well on acoustic scattering for low wave numbers. Among the infinite element formulations, the unconjugated version gave the best results, and it seems to be reasonable to choose $N = 3$ basis functions for the radial shape functions in the infinite elements. No significance was observed by changing the location of the multipole layers in the infinite elements. Moreover, the movement of the artificial boundary farther away from the scatterer increased the accuracy given that the density of the elements was held constant.

As for the FEA, IGA also suffers from the pollution effect at high frequencies. This will always be a problem, and for the higher frequency spectrum, other methods should be used.

When comparing FEA and IGA on the FSI problem on a spherical shell, we observed an advantage for the exact geometry IGA introduce, but also an advantage in its ability for controlling the continuity. The shift of the eigenmodes was greatly reduced using IGA with higher continuity.

When considering the fluid structure interaction problem, we observed that the resolution in the fluid was of greater importance than the resolution in the solid when refinement in the radial direction was considered.

In [10] Burnett concludes by stating that "Structural acoustic modeling with this infinite element is several orders of magnitude faster than the BEM for the same accuracy. Experience with both methods has revealed that the BEM limits practical 3-D problems to low frequencies and simple structures, whereas this infinite element approach can handle the full range of frequencies and structural complexities that are encountered in purely structural analysis, at little additional cost." The majority of implementations in the field uses BEM for handling unbounded domains when FEM is used. It should therefore be an improvement to use the infinite element method instead. To add the contribution from IGA are then considered to be an extra boost in the accuracy for this field.

10.1 Future work

The largest obstacle encountered in this thesis is obviously the memory consumption using a direct solver. To continue to more complex geometries, and perhaps considering higher frequencies, it is essential to get rid of this obstacle. One should therefore turn to iterative techniques where preconditioning should be analyzed.

There are many parameters which deserves a thorough analysis. As mentioned, this includes the location of the layers of the multipole expansion in the infinite element, the number of radial shape functions used in the infinite elements, and the location of the artificial boundary.

All of the implementations used in this thesis have been implemented in MATLAB. For optimization purposes, several parts of the code could have been implemented in `C++`. This especially concerns the implementation of the B-splines.

One of the obvious next steps is to consider local refinement. This improvement will not solve the pollution effect. It is more aimed at refining in areas containing singularities, which is not a problem for scattering problems. However, when constructing the mesh, local refinement can be advantageous for obtaining the most uniform mesh. Moreover, in [11] Gerdes showed that an a-posteriori error estimation for adaptive mesh refinement is necessary for a successful large scale implementation of the infinite element method. As mentioned, several strategies for local mesh refinement has been presented for IGA, which will form a basis for the additional implementations needed to couple IGA with the infinite elements.

The infinite element method is only one of many other methods to handle absorbing boundary conditions. The methods which could reduce the number of elements in the outer fluid domain would be advantageous. Especially the PML method could be interesting to investigate in this regard. That being said, there is still a lot of analysis to be done for the infinite element. For instance, whether it is possible to use a more general coordinate system which could move the artificial boundary closer to the scatterer.

The mentioned evaluation of the radial integrals in the infinite element for the unconjugated version suffers from precision loss for high number n. It should be investigated if it is possible to resolve this problem without having to invoke MuPAD in MATLAB. One needs to simulate on very coarse mesh if the evaluation time of this procedure using MuPAD is of any significance, but more robust algorithms is always appreciated. In [10, p. 2812] Burnett only mentioned this method as an alternative to the use of Fourier transform for evaluation these integrals. It would be interesting to investigate if this method suffers from the same obstacle.

For higher frequencies other techniques must be introduced. One such method considering asymptotic analysis of path approximations is presented in [40]. Moreover, the most used techniques in the BeTSSi community for the involved models is the Kirchhoff approximation. For details consider [41], [42] and [43].

Appendix A

Gauss theorem and its implications

Gauss theorem is given by

$$\oint_{\partial\mathcal{V}} \boldsymbol{\Psi} \cdot \boldsymbol{n} \, \mathrm{d}S = \int_{\mathcal{V}} \nabla \cdot \boldsymbol{\Psi} \, \mathrm{d}\mathcal{V}. \tag{A.1}$$

A.1 Gauss theorem for scalar functions

Let $\{\boldsymbol{e}_i\}_{i=1,2,3}$ be the standard basis for the euclidean space and let $\boldsymbol{\Psi} = \psi \boldsymbol{e}_i$ for some $i \in \{1, 2, 3\}$. Then

$$\begin{aligned} \oint_{\partial\mathcal{V}} \psi \boldsymbol{e}_i \cdot \boldsymbol{n} \, \mathrm{d}S &= \int_{\mathcal{V}} \nabla \cdot \psi \boldsymbol{e}_i \, \mathrm{d}\mathcal{V} \\ \Updownarrow \\ \oint_{\partial\mathcal{V}} \psi n_i \, \mathrm{d}S &= \int_{\mathcal{V}} \partial_{x_i} \psi \, \mathrm{d}\mathcal{V}. \end{aligned}$$

Since this holds for all $i \in \{1, 2, 3\}$, we have three equations which we may combine as follows

$$\begin{aligned} \left\{ \oint_{\partial\mathcal{V}} \psi n_1 \, \mathrm{d}S, \oint_{\partial\mathcal{V}} \psi n_2 \, \mathrm{d}S, \oint_{\partial\mathcal{V}} \psi n_3 \, \mathrm{d}S \right\} &= \left\{ \int_{\mathcal{V}} \partial_{x_1} \psi \, \mathrm{d}\mathcal{V}, \int_{\mathcal{V}} \partial_{x_2} \psi \, \mathrm{d}\mathcal{V}, \int_{\mathcal{V}} \partial_{x_3} \psi \, \mathrm{d}\mathcal{V} \right\} \\ \Updownarrow \\ \oint_{\partial\mathcal{V}} \psi \left\{ n_1, n_2, n_3 \right\} \, \mathrm{d}S &= \int_{\mathcal{V}} \left\{ \partial_{x_1} \psi, \partial_{x_2} \psi, \partial_{x_3} \psi \right\} \, \mathrm{d}\mathcal{V} \\ \Updownarrow \\ \oint_{\partial\mathcal{V}} \psi \boldsymbol{n} \, \mathrm{d}S &= \int_{\mathcal{V}} \nabla \psi \, \mathrm{d}\mathcal{V}. \end{aligned} \tag{A.2}$$

A.2 Green's first identity

Green's first identity is simply Gauss theorem applied to $\boldsymbol{\Psi} = \psi \boldsymbol{\Phi}$ where $\boldsymbol{\Phi} = \nabla \phi$. Insertion of the identity

$$\nabla \cdot (\psi \boldsymbol{\Phi}) = \nabla \psi \cdot \boldsymbol{\Phi} + \psi \nabla \cdot \boldsymbol{\Phi}$$

into Gauss theorem yields

$$\int_\Omega \psi \nabla \cdot \boldsymbol{\Phi} \, \mathrm{d}\mathcal{V} = -\int_\Omega \nabla\psi \cdot \boldsymbol{\Phi} \, \mathrm{d}\mathcal{V} + \int_{\partial\Omega} \psi \boldsymbol{\Phi} \cdot \boldsymbol{n} \, \mathrm{d}S. \tag{A.3}$$

As $\nabla \cdot \boldsymbol{\Phi} = \nabla \cdot \nabla\phi = \Delta\phi$, we arrive at Green's first identity

$$\int_\Omega \psi \Delta\phi \, \mathrm{d}\mathcal{V} = -\int_\Omega \nabla\psi \cdot \nabla\phi \, \mathrm{d}\mathcal{V} + \int_{\partial\Omega} \psi \nabla\phi \cdot \boldsymbol{n} \, \mathrm{d}S. \tag{A.4}$$

Appendix B

Coordinate systems

B.1 General 3D coordinate system

We shall now consider the general 3D coordinate system, which one may reduce to the the cylindrical coordinate system and the spherical coordinate system. Let the coordinate system be defined by

$$\boldsymbol{r} = \begin{bmatrix} x(\xi,\eta,\zeta) \\ y(\xi,\eta,\zeta) \\ z(\xi,\eta,\zeta) \end{bmatrix} : \mathbb{R}^3 \to \mathbb{R}^3, \qquad \begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} \mapsto \boldsymbol{r}(\xi,\eta,\zeta).$$

We shall frequently use the scaling lengths defined by

$$\begin{aligned} h_\xi &= \left\| \frac{\partial \boldsymbol{r}}{\partial \xi} \right\| = \sqrt{\left(\frac{\partial x}{\partial \xi}\right)^2 + \left(\frac{\partial y}{\partial \xi}\right)^2 + \left(\frac{\partial z}{\partial \xi}\right)^2} \\ h_\eta &= \left\| \frac{\partial \boldsymbol{r}}{\partial \eta} \right\| = \sqrt{\left(\frac{\partial x}{\partial \eta}\right)^2 + \left(\frac{\partial y}{\partial \eta}\right)^2 + \left(\frac{\partial z}{\partial \eta}\right)^2} \\ h_\zeta &= \left\| \frac{\partial \boldsymbol{r}}{\partial \zeta} \right\| = \sqrt{\left(\frac{\partial x}{\partial \zeta}\right)^2 + \left(\frac{\partial y}{\partial \zeta}\right)^2 + \left(\frac{\partial z}{\partial \zeta}\right)^2}. \end{aligned} \tag{B.1}$$

The standard basis vectors in this general coordinate system are then

$$\boldsymbol{e}_\xi = \frac{1}{h_\xi}\frac{\partial \boldsymbol{r}}{\partial \xi}, \quad \boldsymbol{e}_\eta = \frac{1}{h_\eta}\frac{\partial \boldsymbol{r}}{\partial \eta}, \quad \boldsymbol{e}_\zeta = \frac{1}{h_\zeta}\frac{\partial \boldsymbol{r}}{\partial \zeta}.$$

We now want to develop an expression for the nabla operator in terms of these unit vectors. In our coordinate system, the nabla operator is defined by

$$\nabla f = \boldsymbol{e}_\xi g_\xi + \boldsymbol{e}_\eta g_\eta + \boldsymbol{e}_\zeta g_\zeta.$$

where g_ξ, g_η and g_ζ are functionals of f which are to be determined.

Moreover, the nabla operator satisfy

$$\mathrm{d}f = \nabla f \cdot \mathrm{d}\boldsymbol{r},$$

where (by the chain rule)

$$\begin{aligned}\mathrm{d}\boldsymbol{r} &= \frac{\partial \boldsymbol{r}}{\partial \xi}\mathrm{d}\xi + \frac{\partial \boldsymbol{r}}{\partial \eta}\mathrm{d}\eta + \frac{\partial \boldsymbol{r}}{\partial \zeta}\mathrm{d}\zeta \\ &= h_\xi \boldsymbol{e}_\xi \mathrm{d}\xi + h_\eta \boldsymbol{e}_\eta \mathrm{d}\eta + h_\zeta \boldsymbol{e}_\zeta \mathrm{d}\zeta.\end{aligned}$$

Using the chain rule once again, we also have

$$\mathrm{d}f = \frac{\partial f}{\partial \xi}\mathrm{d}\xi + \frac{\partial f}{\partial \eta}\mathrm{d}\eta + \frac{\partial f}{\partial \zeta}\mathrm{d}\zeta.$$

Combining the equations for $\mathrm{d}f$ we get the relation

$$\begin{aligned}\frac{\partial f}{\partial \xi}\mathrm{d}\xi + \frac{\partial f}{\partial \eta}\mathrm{d}\eta + \frac{\partial f}{\partial \zeta}\mathrm{d}\zeta &= (\boldsymbol{e}_\xi g_\xi + \boldsymbol{e}_\eta g_\eta + \boldsymbol{e}_\zeta g_\zeta)\cdot(h_\xi \boldsymbol{e}_\xi \mathrm{d}\xi + h_\eta \boldsymbol{e}_\eta \mathrm{d}\eta + h_\zeta \boldsymbol{e}_\zeta \mathrm{d}\zeta) && \text{(B.2)} \\ &= h_\xi(g_\xi + \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\eta g_\eta + \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\zeta g_\zeta)\mathrm{d}\xi && \text{(B.3)} \\ &+ h_\eta(\boldsymbol{e}_\xi \cdot \boldsymbol{e}_\eta g_\xi + g_\eta + \boldsymbol{e}_\eta \cdot \boldsymbol{e}_\zeta g_\zeta)\mathrm{d}\eta && \text{(B.4)} \\ &+ h_\zeta(\boldsymbol{e}_\xi \cdot \boldsymbol{e}_\zeta g_\xi + \boldsymbol{e}_\eta \cdot \boldsymbol{e}_\zeta g_\eta + g_\zeta)\mathrm{d}\zeta && \text{(B.5)}\end{aligned}$$

where we have used the fact that $\boldsymbol{e}_\xi \cdot \boldsymbol{e}_\xi = 1$, $\boldsymbol{e}_\eta \cdot \boldsymbol{e}_\eta = 1$ and $\boldsymbol{e}_\zeta \cdot \boldsymbol{e}_\zeta = 1$. As the coefficients in front of $\mathrm{d}\xi$, $\mathrm{d}\eta$ and $\mathrm{d}\zeta$ must be the same on both side, we get a system of equations given by

$$\begin{aligned} g_\xi + \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\eta g_\eta + \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\zeta g_\zeta &= \frac{1}{h_\xi}\frac{\partial f}{\partial \xi} \\ \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\eta g_\xi + g_\eta + \boldsymbol{e}_\eta \cdot \boldsymbol{e}_\zeta g_\zeta &= \frac{1}{h_\eta}\frac{\partial f}{\partial \eta} \\ \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\zeta g_\xi + \boldsymbol{e}_\eta \cdot \boldsymbol{e}_\zeta g_\eta + g_\zeta &= \frac{1}{h_\zeta}\frac{\partial f}{\partial \zeta}.\end{aligned} \qquad \text{(B.6)}$$

Introducing the notation

$$E_{\xi\eta} = \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\eta, \quad E_{\xi\zeta} = \boldsymbol{e}_\xi \cdot \boldsymbol{e}_\zeta, \quad E_{\eta\zeta} = \boldsymbol{e}_\eta \cdot \boldsymbol{e}_\zeta$$

the system in Equation (B.6) is given by can be written as

$$\begin{bmatrix} 1 & E_{\xi\eta} & E_{\xi\zeta} \\ E_{\xi\eta} & 1 & E_{\eta\zeta} \\ E_{\xi\zeta} & E_{\eta\zeta} & 1 \end{bmatrix} \begin{bmatrix} g_\xi \\ g_\eta \\ g_\zeta \end{bmatrix} = \begin{bmatrix} \frac{1}{h_\xi}\frac{\partial f}{\partial \xi} \\ \frac{1}{h_\eta}\frac{\partial f}{\partial \eta} \\ \frac{1}{h_\zeta}\frac{\partial f}{\partial \zeta} \end{bmatrix}.$$

As the determinant of this system of equation is given by

$$\Delta = 1 - E_{\xi\eta}^2 - E_{\xi\zeta}^2 - E_{\eta\zeta}^2 + 2E_{\xi\eta}E_{\xi\zeta}E_{\eta\zeta}$$

we may write the solution as

$$\begin{bmatrix} g_\xi \\ g_\eta \\ g_\zeta \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} \frac{1}{h_\xi}\frac{\partial f}{\partial \xi} \\ \frac{1}{h_\eta}\frac{\partial f}{\partial \eta} \\ \frac{1}{h_\zeta}\frac{\partial f}{\partial \zeta} \end{bmatrix}$$

where the coefficients a_{ij} are given by the symmetric matrix

$$[a_{ij}] = \frac{1}{\Delta}\begin{bmatrix} 1 - E_{\eta\zeta}^2 & E_{\xi\zeta}E_{\eta\zeta} - E_{\xi\eta} & E_{\xi\eta}E_{\eta\zeta} - E_{\xi\zeta} \\ E_{\xi\zeta}E_{\eta\zeta} - E_{\xi\eta} & 1 - E_{\xi\zeta}^2 & E_{\xi\eta}E_{\xi\zeta} - E_{\eta\zeta} \\ E_{\xi\eta}E_{\eta\zeta} - E_{\xi\zeta} & E_{\xi\eta}E_{\xi\zeta} - E_{\eta\zeta} & 1 - E_{\xi\eta}^2 \end{bmatrix}$$

Let's verify this calculation by finding the nabla operator in the spherical coordinate system. Letting ξ correspond to the azimuthal angle ϕ, η the polar angle θ and ζ the radial direction r, we have the relations

$$\begin{aligned} x &= \zeta \sin\eta \cos\xi \\ y &= \zeta \sin\eta \sin\xi \\ z &= \zeta \cos\eta. \end{aligned}$$

As the spherical coordinate system is an orthogonal coordinate system, we get $E_{\xi\eta} = 0$, $E_{\xi\zeta} = 0$ and $E_{\eta\zeta} = 0$. Thus,

$$g_\xi = \frac{1}{h_\xi}\frac{\partial f}{\partial \xi}, \quad g_\eta = \frac{1}{h_\eta}\frac{\partial f}{\partial \eta}, \quad g_\zeta = \frac{1}{h_\zeta}\frac{\partial f}{\partial \zeta}.$$

By computation, we find

$$h_\xi = \zeta \sin\eta, \quad h_\eta = \zeta, \quad h_\zeta = 1.$$

Thus,

$$\begin{aligned} \nabla f &= \frac{1}{\zeta \sin\eta}\frac{\partial f}{\partial \xi}\boldsymbol{e}_\xi + \frac{1}{\zeta}\frac{\partial f}{\partial \eta}\boldsymbol{e}_\eta + \frac{\partial f}{\partial \zeta}\boldsymbol{e}_\zeta \\ &= \frac{\partial f}{\partial r}\boldsymbol{e}_\mathrm{r} + \frac{1}{r}\frac{\partial f}{\partial \theta}\boldsymbol{e}_\theta + \frac{1}{r\sin\theta}\frac{\partial f}{\partial \phi}\boldsymbol{e}_\phi, \end{aligned}$$

which is the familiar expression for the nabla operator in spherical coordinates.

B.1.1 Extended NURBS coordinate system

In this section we shall explore the possibility to have a more general coordinate system for the infinite elements. All though the prolate spheroidal coordinate system helps reduce the number of elements needed to circumference a long obstacle compared to the spherical coordinate system, it is still not ideal with respect to this problem. It is possible to define a coordinate system such that we get

$$\inf_{\boldsymbol{x}\in\Gamma} \|\boldsymbol{x} - \boldsymbol{y}_0\| = \inf_{\boldsymbol{y}\in\Gamma_\mathrm{a}} \|\boldsymbol{x}_0 - \boldsymbol{y}\| \quad \forall \boldsymbol{x}_0 \in \Gamma, \quad \text{and} \quad \forall \boldsymbol{y}_0 \in \Gamma_\mathrm{a}.$$

That is, we can have the surface Γ_a at a constant distance from the surface Γ (if Γ is convex[1]). However, is it possible to define a infinite element formulation on such a coordinate system? The most intuitive coordinate system would satisfy the property that for all points on Γ_a the normal vector is parallel to the "radial" unit vector for this coordinate system. This would however make the integration in the weak formulation hard as the nabla operator would not be separable, and we may no longer integrate analytically in the "radial" direction. In the following we shall consider another option which results in a separable nabla operator.

Let's build upon the theory developed in Appendix B.1 where we shall assume that the NURBS parametrization at hand has a "radial" parameter ζ, such that ξ and η

[1] If this is not the case, an intermediate convex surface must be inserted between Γ and Γ_a in the NURBS patch.

corresponds to "angular" parameters. This will typically be the case when considering shell objects, where ζ will run through the thickness of the elastic material. We know that at any given value $\bar{\zeta}$ the NURBS solid in 3D reduces to a NURBS surface in 3D. This is in particular true for $\bar{\zeta} = 1$, which will be our artificial boundary Γ_a. We shall assume, without loss of generality, that the NURBS surface Γ_a fully encloses the origin. Finally, we assume the surface Γ_a to be a convex surface. We now want to extend the NURBS parametrization also for $\zeta > 1$. In particular we define (for $\zeta > 1$)

$$\boldsymbol{r} = \begin{bmatrix} x(\xi,\eta,\zeta) \\ y(\xi,\eta,\zeta) \\ z(\xi,\eta,\zeta) \end{bmatrix} : \mathbb{R}^3 \to \mathbb{R}^3, \qquad \begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} \mapsto s(\zeta)\bar{\boldsymbol{r}}(\xi,\eta).$$

where $\bar{\boldsymbol{r}}$ is the NURBS 3D surface at $\zeta = 1$ and the *scaling function* s is strictly increasing and satisfies $s(1) = 1$. Thus, for any $\zeta > 1$ we simply get a scaled version of Γ_a from the origin with the scaling factor $s(\zeta)$.

Continuing on the notation from Appendix B.1, our new coordinate system has the following nice property

$$h_\xi = s(\zeta)\left\|\frac{\partial \bar{\boldsymbol{r}}}{\partial \xi}\right\|, \quad h_\eta = s(\zeta)\left\|\frac{\partial \bar{\boldsymbol{r}}}{\partial \eta}\right\|, \quad h_\zeta = s'(\zeta)\left\|\bar{\boldsymbol{r}}\right\|.$$

From this we observe that the unit vectors $\boldsymbol{e}_\xi$, $\boldsymbol{e}_\eta$ and $\boldsymbol{e}_\zeta$ are all independent of ζ. Moreover, $E_{\xi\eta}$, $E_{\xi\zeta}$ and $E_{\eta\zeta}$ (and thus also the determinant Δ) will also be independent of ζ. The resulting nabla operator will then be separable, which we will take advantage of. In this context we define

$$\bar{h}_\xi = \left\|\frac{\partial \bar{\boldsymbol{r}}}{\partial \xi}\right\|, \quad \bar{h}_\eta = \left\|\frac{\partial \bar{\boldsymbol{r}}}{\partial \eta}\right\|, \quad \bar{h}_\zeta = \left\|\bar{\boldsymbol{r}}\right\|$$

such that $h_\xi = s(\zeta)\bar{h}_\xi$, $h_\eta = s(\zeta)\bar{h}_\eta$, $h_\zeta = s'(\zeta)\bar{h}_\zeta$. We may now easy see that the nabla operator is separable as it is now given by (for $\zeta > 1$)

$$\begin{aligned}
\nabla &= \left(\frac{a_{11}}{s(\zeta)\bar{h}_\xi}\frac{\partial}{\partial \xi} + \frac{a_{12}}{s(\zeta)\bar{h}_\eta}\frac{\partial}{\partial \eta} + \frac{a_{13}}{s'(\zeta)\bar{h}_\zeta}\frac{\partial}{\partial \zeta}\right)\boldsymbol{e}_\xi \\
&+ \left(\frac{a_{21}}{s(\zeta)\bar{h}_\xi}\frac{\partial}{\partial \xi} + \frac{a_{22}}{s(\zeta)\bar{h}_\eta}\frac{\partial}{\partial \eta} + \frac{a_{23}}{s'(\zeta)\bar{h}_\zeta}\frac{\partial}{\partial \zeta}\right)\boldsymbol{e}_\eta \\
&+ \left(\frac{a_{31}}{s(\zeta)\bar{h}_\xi}\frac{\partial}{\partial \xi} + \frac{a_{32}}{s(\zeta)\bar{h}_\eta}\frac{\partial}{\partial \eta} + \frac{a_{33}}{s'(\zeta)\bar{h}_\zeta}\frac{\partial}{\partial \zeta}\right)\boldsymbol{e}_\zeta \\
&= \frac{1}{s(\zeta)\bar{h}_\xi}\left(a_{11}\boldsymbol{e}_\xi + a_{21}\boldsymbol{e}_\eta + a_{31}\boldsymbol{e}_\zeta\right)\frac{\partial}{\partial \xi} \\
&+ \frac{1}{s(\zeta)\bar{h}_\eta}\left(a_{12}\boldsymbol{e}_\xi + a_{22}\boldsymbol{e}_\eta + a_{32}\boldsymbol{e}_\zeta\right)\frac{\partial}{\partial \eta} \\
&+ \frac{1}{s'(\zeta)\bar{h}_\zeta}\left(a_{13}\boldsymbol{e}_\xi + a_{23}\boldsymbol{e}_\eta + a_{33}\boldsymbol{e}_\zeta\right)\frac{\partial}{\partial \zeta} \\
&= \frac{1}{s(\zeta)}\nabla_S + \frac{1}{s'(\zeta)}\boldsymbol{a}\frac{\partial}{\partial \zeta}
\end{aligned}$$

where

$$\nabla_S = \frac{1}{\bar{h}_\xi}\left(a_{11}\boldsymbol{e}_\xi + a_{21}\boldsymbol{e}_\eta + a_{31}\boldsymbol{e}_\zeta\right)\frac{\partial}{\partial \xi} + \frac{1}{\bar{h}_\eta}\left(a_{12}\boldsymbol{e}_\xi + a_{22}\boldsymbol{e}_\eta + a_{32}\boldsymbol{e}_\zeta\right)\frac{\partial}{\partial \eta}$$

and

$$\boldsymbol{a} = \frac{1}{\bar{h}_\zeta}\left(a_{13}\boldsymbol{e}_\xi + a_{23}\boldsymbol{e}_\eta + a_{33}\boldsymbol{e}_\zeta\right).$$

We note that (for any normal vector $\boldsymbol{n}$ on the surface Γ_a) we have

$$\boldsymbol{a}\cdot\boldsymbol{n} = \frac{a_{33}}{\bar{h}_\zeta}\boldsymbol{e}_\zeta\cdot\boldsymbol{n}. \tag{B.7}$$

and

$$\|\boldsymbol{a}\| = \frac{1}{\bar{h}_\zeta}\sqrt{a_{33}}.$$

Moreover, we note that in the case $\boldsymbol{e}_\xi \perp \boldsymbol{e}_\zeta$ and $\boldsymbol{e}_\eta \perp \boldsymbol{e}_\zeta$ (for example when Γ_a is a sphere), this vector reduces to $\boldsymbol{a} = a_{33}\boldsymbol{e}_\zeta/\bar{h}_\zeta$.

Using the expression for the Jacobian matrix in Equation (4.20) we get

$$|\det(\boldsymbol{J})| = [s(\zeta)]^2 s'(\zeta)\left\|\frac{\partial\bar{\boldsymbol{r}}}{\partial\xi}\times\frac{\partial\bar{\boldsymbol{r}}}{\partial\eta}\right\|\bar{h}_\zeta\boldsymbol{n}\cdot\boldsymbol{e}_\zeta.$$

We may therefore write

$$\mathrm{d}\Omega = [s(\zeta)]^2 s'(\zeta)\bar{h}_\zeta \mathrm{d}S\mathrm{d}\zeta$$

where

$$\mathrm{d}S = \left\|\frac{\partial\bar{\boldsymbol{r}}}{\partial\xi}\times\frac{\partial\bar{\boldsymbol{r}}}{\partial\eta}\right\|\boldsymbol{n}\cdot\boldsymbol{e}_\zeta\mathrm{d}\xi\mathrm{d}\eta.$$

We also note that a surface where $\zeta = \text{const}$, the surface element is

$$\mathrm{d}\Gamma = [s(\zeta)]^2\left\|\frac{\partial\bar{\boldsymbol{r}}}{\partial\xi}\times\frac{\partial\bar{\boldsymbol{r}}}{\partial\eta}\right\|\mathrm{d}\xi\mathrm{d}\eta = \frac{[s(\zeta)]^2}{\boldsymbol{n}\cdot\boldsymbol{e}_\zeta}\mathrm{d}S.$$

Hence, both ∇_S and $\mathrm{d}S$ are independent of ζ which enables us to separate the integral in the weak form.

B.1.2 Weak formulation for infinite elements

We shall use the most natural choice of the scaling function, namely $s(\zeta) = \zeta$. This simplifications result in the following

$$\nabla = \frac{1}{\zeta}\nabla_S + \frac{\boldsymbol{e}_\zeta}{R_\mathrm{a}}\frac{\partial}{\partial\zeta}, \quad \text{where} \quad \nabla_S = \frac{\boldsymbol{e}_\xi}{\bar{h}_\xi}\frac{\partial}{\partial\xi} + \frac{\boldsymbol{e}_\eta}{\bar{h}_\eta}\frac{\partial}{\partial\eta}$$

and

$$\mathrm{d}\Omega = \zeta^2 R_\mathrm{a}\mathrm{d}S, \qquad \mathrm{d}\Gamma = \zeta^2\mathrm{d}S, \quad \text{where} \quad \mathrm{d}S = \left\|\frac{\partial\bar{\boldsymbol{r}}}{\partial\xi}\times\frac{\partial\bar{\boldsymbol{r}}}{\partial\eta}\right\|\mathrm{d}\xi\mathrm{d}\eta.$$

In order to use the extended NURBS coordinate system, we have to modify the "radial shape functions" which Ihlenburg presents on the form

$$\phi_n(r) = \frac{\mathrm{e}^{\mathrm{i}kr}}{r^n}, \quad n = 1,\ldots,N.$$

We can not simply replace r with ζ here, as the latter is dimensionless and invariant of the size of the object. Note that the wave number k has dimension m^{-1}, so we must

scale ζ by a parameter with dimension m. To get correspondence with these functions in spherical coordinate system, we have to scale with the "radius" of the artificial boundary, R_a. This quantity must be independent of the angular parameters in order to make separable functions (the average distance from the origin to the surface Γ_a may be the most natural choice). That is,

$$\tilde{\phi}(\zeta) = \frac{\mathrm{e}^{\mathrm{i}R_\mathrm{a}k\zeta}}{(R_\mathrm{a}\zeta)^n}, \quad n = 1, \ldots, N. \tag{B.8}$$

The weak formulating takes the form: For all $v \in V_1$, find $u_h^N \in V_1$ such that

$$b_{uc}(u_h^N, v) = \langle g, v \rangle_\Gamma,$$

where

$$\begin{aligned} b_{uc}(v,u) &= \lim_{\gamma\to\infty} \left(\int_{\Omega_\gamma} (\nabla v \nabla u - k^2 v u)\,\mathrm{d}\Omega - \int_{S_\gamma} v\partial_n u\,\mathrm{d}\Gamma \right), \\ \langle g, v\rangle_\Gamma &= \int_\Gamma g v\,\mathrm{d}\Gamma. \end{aligned} \tag{B.9}$$

Here, S_γ is the surface where $\zeta = \gamma$ and we can then recover the full domain by letting $\gamma \to \infty$. For the domain outside the artificial boundary ($\zeta = 1$) we consider trial and test functions of the form

$$u = \frac{\mathrm{e}^{\mathrm{i}R_\mathrm{a}k\zeta}}{\zeta^m} f_m(\xi,\eta), \quad v = \frac{\mathrm{e}^{\mathrm{i}R_\mathrm{a}k\zeta}}{\zeta^n} f_n(\xi,\eta),$$

where we have baked the constants R_a^n and R_a^m from the radial shape functions in Equation (B.8), into f_n and f_m respectively. As these functions are separable, we get

$$\begin{aligned} \nabla v \cdot \nabla u &= \left(\frac{1}{\zeta}\nabla_S v + \frac{\boldsymbol{e}_\zeta}{R_\mathrm{a}}\frac{\partial v}{\partial \zeta} \right) \cdot \left(\frac{1}{\zeta}\nabla_S u + \frac{\boldsymbol{e}_\zeta}{R_\mathrm{a}}\frac{\partial u}{\partial \zeta} \right) \\ &= \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{\zeta^{m+n+2}} \nabla_S f_m \cdot \nabla_S f_n + \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{R_\mathrm{a}^2\zeta^{m+n}} \left(\mathrm{i}R_\mathrm{a}k - \frac{n}{\zeta} \right) \left(\mathrm{i}R_\mathrm{a}k - \frac{m}{\zeta} \right) f_m f_n. \end{aligned}$$

and

$$\begin{aligned} v\partial_n u = v\nabla u \cdot \boldsymbol{n} &= v \left(\frac{1}{\zeta}\nabla_S u + \frac{\boldsymbol{e}_\zeta}{R_\mathrm{a}}\frac{\partial u}{\partial \zeta} \right) \cdot \boldsymbol{e}_\zeta = \frac{v}{R_\mathrm{a}}\frac{\partial u}{\partial \zeta} \\ &= \left(\mathrm{i}R_\mathrm{a}k - \frac{m}{\zeta} \right) \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{R_\mathrm{a}\zeta^{m+n}} f_n f_m. \end{aligned}$$

Consider first the boundary integral at S_γ

$$\begin{aligned} \int_{S_\gamma} v\partial_n u\,\mathrm{d}\Gamma &= \int_{S_\gamma} \left(\mathrm{i}R_\mathrm{a}k - \frac{m}{\zeta} \right) \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{R_\mathrm{a}\zeta^{m+n}} f_n f_m \zeta^2\,\mathrm{d}S \\ &= \left(\mathrm{i}R_\mathrm{a}k - \frac{m}{\gamma} \right) \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma}}{R_\mathrm{a}\gamma^{m+n-2}} \int_{\Gamma_\mathrm{a}} f_n f_m\,\mathrm{d}S \end{aligned}$$

As $m+n>1$ all terms of order $\mathcal{O}(\gamma^{-(m+n-1)})$ vanish in the limit $\gamma\to\infty$, such that the integral reduces to

$$\int_{S_\gamma} v\partial_n u\,\mathrm{d}\Gamma = \frac{\mathrm{i}k\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma}}{\gamma^{m+n-2}}\int_{\Gamma_\mathrm{a}} f_n f_m\,\mathrm{d}S$$

Combining all of this into Equation (B.9) yields

$$\begin{aligned} b_{uc}(v,u) = \lim_{\gamma\to\infty}\Bigg\{ & J_{mn}\int_1^\gamma \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{\zeta^{m+n}}\,\mathrm{d}\zeta \\ & + I_{mn}\int_1^\gamma \frac{\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}}{\zeta^{m+n}}\left[-2(R_\mathrm{a}k\zeta)^2 - \mathrm{i}R_\mathrm{a}k\zeta(n+m)+nm\right]\,\mathrm{d}\zeta \\ & - I_{mn}\frac{\mathrm{i}R_\mathrm{a}k\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma}}{\gamma^{m+n-2}}\Bigg\} \end{aligned}$$

where

$$I_{mn} = \int_{\Gamma_\mathrm{a}} \frac{1}{R_\mathrm{a}} f_m f_n\,\mathrm{d}S, \qquad J_{mn} = \int_{\Gamma_\mathrm{a}} R_\mathrm{a}\nabla_S f_m\cdot\nabla_S f_n\,\mathrm{d}S \tag{B.10}$$

Thus, we get the same radial integrals as before. It remains to consider the case $n=m=1$. As

$$\begin{aligned} L &= \lim_{\gamma\to\infty}\left[-2(R_\mathrm{a}k)^2 I_{11}\int_1^\gamma \mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\zeta}\,\mathrm{d}\zeta - \mathrm{i}R_\mathrm{a}kI_{11}\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma}\right] \\ &= \lim_{\gamma\to\infty}\left[\mathrm{i}R_\mathrm{a}kI_{11}(\,\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma} - \mathrm{e}^{2\mathrm{i}R_\mathrm{a}k}\,) - \mathrm{i}R_\mathrm{a}kI_{11}\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k\gamma}\right] \\ &= -\mathrm{i}R_\mathrm{a}kI_{11}\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k}. \end{aligned}$$

we get the bilinear form

$$b_{uc}(v,u) = J_{mn}E_{m+n} + I_{mn}[-2(R_\mathrm{a}k)^2E_{m+n-2} - \mathrm{i}R_\mathrm{a}k(n+m)E_{m+n-1} + nmE_{m+n}]$$

where $E_n = E_n(-2\mathrm{i}R_\mathrm{a}k)$. And for the special case $n=m=1$ we get

$$b_{uc}(v,u) = J_{11}E_2 + I_{11}[-2\mathrm{i}R_\mathrm{a}kE_1 + E_2 - \mathrm{i}R_\mathrm{a}k\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k}]$$

Let

$$f_n = \sum_{A\in\boldsymbol{\eta}_\mathrm{a}} R_A c_{nA}, \quad f_m = \sum_{B\in\boldsymbol{\eta}_\mathrm{a}} R_B d_{mB}$$

where $\boldsymbol{\eta}_\mathrm{a}$ is the set containing the indices of all the basis functions that are non-zero on Γ_a. Using the bilinearity of b_{uc} we obtain (for $\zeta>1$)

$$\sum_{A\in\boldsymbol{\eta}_\mathrm{a}}\sum_{n=1}^{N} c_{nA}\left(\sum_{B\in\boldsymbol{\eta}_\mathrm{a}}\sum_{m=1}^{N} d_{nA}b_{uc}(R_A,R_B)\right) = 0.$$

As the surface integrals in Equation (B.10) will now be independent of the indices m and n we now get (for $n+m>1$)

$$b_{uc}(R_A,R_B) = J_{AB}E_{m+n} + I_{AB}[-2(R_\mathrm{a}k)^2E_{m+n-2} - \mathrm{i}R_\mathrm{a}k(n+m)E_{m+n-1} + nmE_{m+n}]$$

and in the case $n=m=1$

$$b_{uc}(R_A,R_B) = J_{AB}E_2 + I_{AB}[-2\mathrm{i}R_\mathrm{a}kE_1 + E_2 - \mathrm{i}R_\mathrm{a}k\mathrm{e}^{2\mathrm{i}R_\mathrm{a}k}]$$

where we have redefined the surface integral notations to be

$$I_{AB} = \int_{\Gamma_a} \frac{1}{R_a} R_A R_B \, \mathrm{d}S, \qquad J_{AB} = \int_{\Gamma_a} R_a \nabla_S R_A \cdot \nabla_S R_B \, \mathrm{d}S$$

This formulation has proven to work for a sphere when the "radius" R_a for the artificial boundary is constant. In [10] Burnett writes that the scattered pressure field exterior to a spherical Γ_a may be written as

$$p = \frac{\mathrm{e}^{\mathrm{i}kr}}{r} \sum_{n=0}^{\infty} \frac{F_n(\theta, \phi; k)}{r^n}$$

where we have adjusted the sign in the exponential function to match the convention in [3] and this thesis. It has been proven that this series has nice convergence properties and the functions $F_n(\theta, \phi; k)$ for $n > 0$ may be determined by the recursive relation

$$2\mathrm{i}knF_n = \left[n(n-1) + \frac{1}{\sin\theta} \frac{\partial}{\partial\theta} \left(\sin\theta \frac{\partial}{\partial\theta} \right) + \frac{1}{\sin^2\theta} \frac{\partial^2}{\partial\phi^2} \right] F_{n-1}.$$

Here F_0 is simply the radiation pattern which in this thesis has the name F_k.

Correspondingly, in the prolate spheroidal coordinate system the series

$$p = \frac{\mathrm{e}^{\mathrm{i}kr}}{r} \sum_{n=0}^{\infty} \frac{G_n(\theta, \phi; k)}{r^n}$$

also converges absolutely and uniformly for any point outside Γ_a with a six-term recursion formula for the functions $G_n(\theta, \phi; k)$.

To prove the convergence of such series for the coordinate system discussed in this section is outside the scope of this thesis. Due to the lacking mathematical foundation of this infinite element formulation and the time constraint of the thesis, this formulation was not analyzed further in this thesis.

The motivation of introducing the extended NURBS coordinate system is to be able to control the artificial boundary Γ_a. This allows us to control the size of the fluid domain which would in turn enable us to reduce redundant fluid elements. Ihlenburg uses a prolate spheroid as the artificial boundary around his "Mock shell" (depicted in Figure B.1) in [3]. As the "Mock shell" is a convex surface, one could have a uniform thickness of the fluid around the shell by the extended NURBS coordinate system, and thus reducing the fluid computational domain to an ideal size.

B.2 The cylindrical coordinate system

The cylindrical coordinate system is simply an extension of the polar coordinate system, which from standard Cartesian coordinates is given by the transformation

$$x = r\cos\theta$$
$$y = r\sin\theta$$

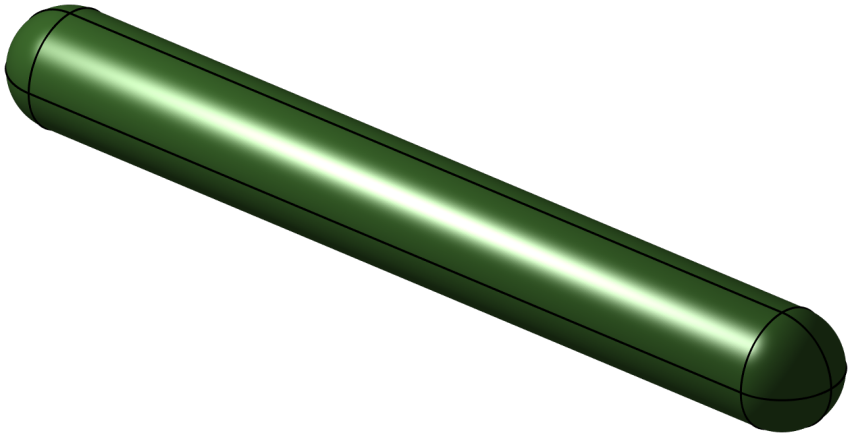


Figure B.1: The mock shell.

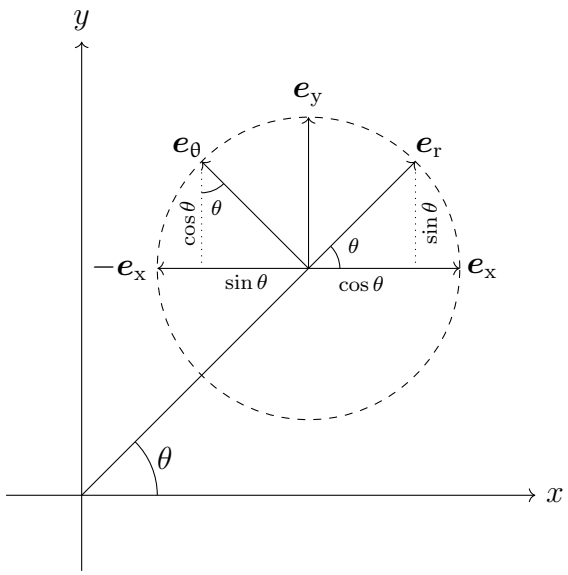


Figure B.2: Decomposition of the standard unit vectors in the polar coordinate system to the standard unit vectors in the Cartesian coordinate system.

such that

$$\begin{aligned} r &= \sqrt{x^2 + y^2} \\ \theta &= \operatorname{atan2}(y, x) \end{aligned}$$

where

$$\operatorname{atan2}(y,x) = \begin{cases} \arctan(\frac{y}{x}) & \text{if } x > 0 \\ \arctan(\frac{y}{x}) + \pi & \text{if } x < 0 \text{ and } y \geq 0 \\ \arctan(\frac{y}{x}) - \pi & \text{if } x < 0 \text{ and } y < 0 \\ \frac{\pi}{2} & \text{if } x = 0 \text{ and } y > 0 \\ -\frac{\pi}{2} & \text{if } x = 0 \text{ and } y < 0 \\ \text{undefined} & \text{if } x = 0 \text{ and } y = 0 \end{cases}.$$

By this we find

$$\begin{aligned} \frac{\partial r}{\partial x} &= \cos\theta, & \frac{\partial r}{\partial y} &= \sin\theta \\ \frac{\partial \theta}{\partial x} &= -\frac{1}{r}\sin\theta, & \frac{\partial \theta}{\partial y} &= \frac{1}{r}\cos\theta. \end{aligned}$$

So for a scalar valued function ψ we get (using the chain rule)

$$\begin{aligned} \frac{\partial \psi}{\partial x} &= \frac{\partial \psi}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial x} = \cos\theta\frac{\partial \psi}{\partial r} - \frac{1}{r}\sin\theta\frac{\partial \psi}{\partial \theta} \\ \frac{\partial \psi}{\partial y} &= \frac{\partial \psi}{\partial r}\frac{\partial r}{\partial y} + \frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial y} = \sin\theta\frac{\partial \psi}{\partial r} + \frac{1}{r}\cos\theta\frac{\partial \psi}{\partial \theta} \end{aligned} \tag{B.11}$$

Moreover, by decomposing the standard unit vectors in the polar coordinate system to the standard unit vectors in the Cartesian coordinate system (cf. Figure B.2) we get

$$\begin{aligned} \boldsymbol{e}_{\mathrm{r}} &= \cos\theta\boldsymbol{e}_{\mathrm{x}} + \sin\theta\boldsymbol{e}_{\mathrm{y}} \\ \boldsymbol{e}_{\theta} &= -\sin\theta\boldsymbol{e}_{\mathrm{x}} + \cos\theta\boldsymbol{e}_{\mathrm{y}}. \end{aligned}$$

Hence, for any vector valued function in two dimensions

$$\boldsymbol{\Psi} = \Psi_{\mathrm{x}}\boldsymbol{e}_{\mathrm{x}} + \Psi_{\mathrm{y}}\boldsymbol{e}_{\mathrm{y}} = \Psi_{\mathrm{r}}\boldsymbol{e}_{\mathrm{r}} + \Psi_{\theta}\boldsymbol{e}_{\theta}$$

we get the relations (by comparing each component)

$$\begin{aligned} \Psi_{\mathrm{x}} &= \Psi_{\mathrm{r}}\cos\theta - \Psi_{\theta}\sin\theta \\ \Psi_{\mathrm{y}} &= \Psi_{\mathrm{r}}\sin\theta + \Psi_{\theta}\cos\theta. \end{aligned} \tag{B.12}$$

By a corresponding argument, we have

$$\begin{aligned} \boldsymbol{e}_{\mathrm{x}} &= \cos\theta\boldsymbol{e}_{\mathrm{r}} - \sin\theta\boldsymbol{e}_{\theta} \\ \boldsymbol{e}_{\mathrm{y}} &= \sin\theta\boldsymbol{e}_{\mathrm{r}} + \cos\theta\boldsymbol{e}_{\theta} \end{aligned} \tag{B.13}$$

such that

$$\begin{aligned} \Psi_{\mathrm{r}} &= \Psi_{\mathrm{x}}\cos\theta + \Psi_{\mathrm{y}}\sin\theta \\ \Psi_{\theta} &= -\Psi_{\mathrm{x}}\sin\theta + \Psi_{\mathrm{y}}\cos\theta. \end{aligned}$$

All of these relation also holds for cylindrical coordinates as the z variable is the same for both coordinate systems.

Using Equation (B.11), Equation (B.12) and Equation (B.13) we get

$$\nabla\psi = \frac{\partial\psi}{\partial x}\boldsymbol{e}_{\mathrm{x}} + \frac{\partial\psi}{\partial y}\boldsymbol{e}_{\mathrm{y}} + \frac{\partial\psi}{\partial z}\boldsymbol{e}_{\mathrm{z}} = \frac{\partial\psi}{\partial r}\boldsymbol{e}_{\mathrm{r}} + \frac{1}{r}\frac{\partial\psi}{\partial\theta}\boldsymbol{e}_{\theta} + \frac{\partial\psi}{\partial z}\boldsymbol{e}_{\mathrm{z}} \tag{B.14}$$

$$\Delta\psi = \frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} + \frac{\partial^2\psi}{\partial z^2} = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\psi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2\psi}{\partial\theta^2} + \frac{\partial^2\psi}{\partial z^2} \tag{B.15}$$

$$\nabla\cdot\boldsymbol{\Psi} = \frac{\partial\Psi_{\mathrm{x}}}{\partial x} + \frac{\partial\Psi_{\mathrm{y}}}{\partial y} + \frac{\partial\Psi_{\mathrm{z}}}{\partial z} = \frac{1}{r}\frac{\partial(r\Psi_{\mathrm{r}})}{\partial r} + \frac{1}{r}\frac{\partial\Psi_{\theta}}{\partial\theta} + \frac{\partial\Psi_{\mathrm{z}}}{\partial z} \tag{B.16}$$

$$\Delta\boldsymbol{\Psi} = \left(\Delta\Psi_{\mathrm{r}} - \frac{1}{r^2}\Psi_{\mathrm{r}} - \frac{2}{r^2}\frac{\partial\Psi_{\theta}}{\partial\theta}\right)\boldsymbol{e}_{\mathrm{r}} + \left(\Delta\Psi_{\theta} - \frac{1}{r^2}\Psi_{\theta} + \frac{2}{r^2}\frac{\partial\Psi_{\mathrm{r}}}{\partial\theta}\right)\boldsymbol{e}_{\theta} + \Delta\Psi_{\mathrm{z}}\boldsymbol{e}_{\mathrm{z}} \tag{B.17}$$

For axisymmetric problems, it is also convenient to write the stress field and the strain field in terms of polar coordinates. Recall that

$$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{33}\\ \sigma_{23}\\ \sigma_{13}\\ \sigma_{12}\end{bmatrix} = \boldsymbol{C}\begin{bmatrix}\varepsilon_{11}\\ \varepsilon_{22}\\ \varepsilon_{33}\\ 2\varepsilon_{23}\\ 2\varepsilon_{13}\\ 2\varepsilon_{12}\end{bmatrix} = \begin{bmatrix}2\mu+\lambda & \lambda & \lambda & 0 & 0 & 0\\ \lambda & 2\mu+\lambda & \lambda & 0 & 0 & 0\\ \lambda & \lambda & 2\mu+\lambda & 0 & 0 & 0\\ 0 & 0 & 0 & \mu & 0 & 0\\ 0 & 0 & 0 & 0 & \mu & 0\\ 0 & 0 & 0 & 0 & 0 & \mu\end{bmatrix}\begin{bmatrix}\varepsilon_{11}\\ \varepsilon_{22}\\ \varepsilon_{33}\\ 2\varepsilon_{23}\\ 2\varepsilon_{13}\\ 2\varepsilon_{12}\end{bmatrix}.$$

such that if the stress field is given, we may invert the elasticity matrix to find the strain field. We note that

$$\boldsymbol{C}^{-1} = \frac{1}{E}\begin{bmatrix}1 & -\nu & -\nu & 0 & 0 & 0\\ -\nu & 1 & -\nu & 0 & 0 & 0\\ -\nu & -\nu & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & \frac{1}{\mu} & 0 & 0\\ 0 & 0 & 0 & 0 & \frac{1}{\mu} & 0\\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\mu}\end{bmatrix}.$$

In [44, p. 19] we find the transformation formula for the stress tensor from an arbitrary coordinate system to another. If $\boldsymbol{e}_i$ and $\boldsymbol{e}'_i$ represents the basis vectors of these two coordinate systems and the stress field is known in the first coordinate system, then the stress field in terms of the second coordinate system is found by

$$\sigma'_{ij} = \alpha_{ik}\alpha_{jl}\sigma_{kl}$$

where

$$\alpha_{ij} = \cos(\boldsymbol{e}'_i, \boldsymbol{e}_j) = \boldsymbol{e}'_i\cdot\boldsymbol{e}_j$$

represents the cosine of the angle between the axis corresponding to the vectors $\boldsymbol{e}_i$ and $\boldsymbol{e}'_i$, respectively. Letting $\boldsymbol{e}'_1 = \boldsymbol{e}_{\mathrm{r}}$, $\boldsymbol{e}'_2 = \boldsymbol{e}_{\mathrm{r}}$ and $\boldsymbol{e}'_3 = \boldsymbol{e}_{\mathrm{z}}$ (the basis vectors in cylindrical coordinates), and $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ the standard basis vectors in Cartesian coordinates, we find (using Equation (B.13))

$$[\alpha_{ij}] = \begin{bmatrix}\cos\theta & \sin\theta & 0\\ -\sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}.$$

This yields the following relations

$$\begin{aligned}
\sigma_{\mathrm{rr}} &= \sigma_{11}\cos^2\theta + \sigma_{22}\sin^2\theta + \sigma_{12}\sin(2\theta)\\
\sigma_{\theta\theta} &= \sigma_{11}\sin^2\theta + \sigma_{22}\cos^2\theta - \sigma_{12}\sin(2\theta)\\
\sigma_{\mathrm{zz}} &= \sigma_{33}\\
\sigma_{\theta\mathrm{z}} &= \sigma_{23}\cos\theta - \sigma_{13}\sin\theta\\
\sigma_{\mathrm{rz}} &= \sigma_{23}\sin\theta + \sigma_{13}\cos\theta\\
\sigma_{\mathrm{r}\theta} &= -\frac{1}{2}\sigma_{11}\sin(2\theta) + \frac{1}{2}\sigma_{22}\sin(2\theta) + \sigma_{12}\cos(2\theta)
\end{aligned}$$

The inverse relations are found to be

$$\begin{aligned}
\sigma_{11} &= \sigma_{\mathrm{rr}}\cos^2\theta + \sigma_{\theta\theta}\sin^2\theta - \sigma_{\mathrm{r}\theta}\sin(2\theta)\\
\sigma_{22} &= \sigma_{\mathrm{rr}}\sin^2\theta + \sigma_{\theta\theta}\cos^2\theta + \sigma_{\mathrm{r}\theta}\sin(2\theta)\\
\sigma_{33} &= \sigma_{\mathrm{zz}}\\
\sigma_{23} &= \sigma_{\theta\mathrm{z}}\cos\theta + \sigma_{\mathrm{rz}}\sin\theta\\
\sigma_{13} &= -\sigma_{\theta\mathrm{z}}\sin\theta + \sigma_{\mathrm{rz}}\cos\theta\\
\sigma_{12} &= \frac{1}{2}\sin(2\theta)\sigma_{\mathrm{rr}} - \frac{1}{2}\sin(2\theta)\sigma_{\theta\theta} - \cos(2\theta)\sigma_{\mathrm{r}\theta}.
\end{aligned}$$

Moreover, we have

$$\begin{bmatrix}\sigma_{\mathrm{rr}}\\ \sigma_{\theta\theta}\\ \sigma_{\mathrm{zz}}\\ \sigma_{\theta\mathrm{z}}\\ \sigma_{\mathrm{rz}}\\ \sigma_{\mathrm{r}\theta}\end{bmatrix} = \boldsymbol{C}\begin{bmatrix}\varepsilon_{\mathrm{rr}}\\ \varepsilon_{\theta\theta}\\ \varepsilon_{\mathrm{zz}}\\ 2\varepsilon_{\theta\mathrm{z}}\\ 2\varepsilon_{\mathrm{rz}}\\ 2\varepsilon_{\mathrm{r}\theta}\end{bmatrix}, \tag{B.18}$$

where (see [45] for details)

$$\begin{aligned}
\varepsilon_{\mathrm{rr}} &= \frac{\partial u_{\mathrm{r}}}{\partial r}\\
\varepsilon_{\theta\theta} &= \frac{1}{r}\left(\frac{\partial u_{\theta}}{\partial\theta} + u_{\mathrm{r}}\right)\\
\varepsilon_{\mathrm{zz}} &= \frac{\partial u_z}{\partial z}\\
\varepsilon_{\theta\mathrm{z}} &= \frac{1}{2}\left(\frac{\partial u_{\theta}}{\partial z} + \frac{1}{r}\frac{\partial u_z}{\partial\theta}\right)\\
\varepsilon_{\mathrm{rz}} &= \frac{1}{2}\left(\frac{\partial u_{\mathrm{r}}}{\partial z} + \frac{\partial u_z}{\partial r}\right)\\
\varepsilon_{\mathrm{r}\theta} &= \frac{1}{2}\left(\frac{1}{r}\frac{\partial u_{\mathrm{r}}}{\partial\theta} + \frac{\partial u_{\theta}}{\partial r} - \frac{u_{\theta}}{r}\right).
\end{aligned} \tag{B.19}$$

B.3 The spherical coordinate system

There are several conventions for defining the spherical coordinate system, but we shall here follow the ISO standard (International Organization for Standardization), which is

the convention often used in physics. The reason why the mathematical convention of the spherical coordinate system is not used (where the angle θ is preserved in the xy-plane as in polar coordinates) is because of the existing literature on the subject of this thesis (in particular [3]). The spherical coordinate system is defined by the transformation (from the standard Cartesian coordinates)

$$\begin{aligned} x &= r\sin\theta\cos\phi \\ y &= r\sin\theta\sin\phi \\ z &= r\cos\theta \end{aligned}$$

such that

$$\begin{aligned} r &= \sqrt{x^2+y^2+z^2} \\ \theta &= \arccos\left(\frac{z}{\sqrt{x^2+y^2+z^2}}\right) \\ \phi &= \operatorname{atan2}(y,x). \end{aligned}$$

By this we find

$$\begin{aligned} &\frac{\partial r}{\partial x} = \sin\theta\cos\phi, && \frac{\partial r}{\partial y} = \sin\theta\sin\phi, && \frac{\partial r}{\partial z} = \cos\theta \\ &\frac{\partial \theta}{\partial x} = \frac{1}{r}\cos\theta\cos\phi, && \frac{\partial \theta}{\partial y} = \frac{1}{r}\cos\theta\sin\phi, && \frac{\partial \theta}{\partial z} = -\frac{1}{r}\sin\theta \\ &\frac{\partial \phi}{\partial x} = -\frac{1}{r}\frac{\sin\phi}{\sin\theta}, && \frac{\partial \phi}{\partial y} = \frac{1}{r}\frac{\cos\phi}{\sin\theta}, && \frac{\partial \phi}{\partial z} = 0. \end{aligned}$$

So for a scalar valued function ψ we get (using the chain rule)

$$\begin{aligned} \frac{\partial \psi}{\partial x} &= \frac{\partial \psi}{\partial r}\frac{\partial r}{\partial x} + \frac{\partial \psi}{\partial \phi}\frac{\partial \phi}{\partial x} + \frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial x} = \sin\theta\cos\phi\frac{\partial \psi}{\partial r} - \frac{1}{r}\frac{\sin\phi}{\sin\theta}\frac{\partial \psi}{\partial \phi} + \frac{1}{r}\cos\theta\cos\phi\frac{\partial \psi}{\partial \theta} \\ \frac{\partial \psi}{\partial y} &= \frac{\partial \psi}{\partial r}\frac{\partial r}{\partial y} + \frac{1}{r}\frac{\cos\phi}{\sin\theta}\frac{\partial \psi}{\partial \phi} + \frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial y} = \sin\theta\cos\phi\frac{\partial \psi}{\partial r} + \frac{\partial \psi}{\partial \phi}\frac{\partial \phi}{\partial y} + \frac{1}{r}\cos\theta\sin\phi\frac{\partial \psi}{\partial \theta} \\ \frac{\partial \psi}{\partial z} &= \frac{\partial \psi}{\partial r}\frac{\partial r}{\partial z} + \frac{\partial \psi}{\partial \phi}\frac{\partial \phi}{\partial z} + \frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial z} = \cos\theta\frac{\partial \psi}{\partial r} - \frac{1}{r}\sin\theta\frac{\partial \psi}{\partial \theta}\frac{\partial \theta}{\partial z} \end{aligned} \tag{B.20}$$

Moreover, by decomposing the standard unit vectors in the polar coordinate system to the standard unit vectors in the Cartesian coordinate system we get

$$\begin{aligned} \boldsymbol{e}_{\mathrm{r}} &= \sin\theta\cos\phi\boldsymbol{e}_{\mathrm{x}} + \sin\theta\sin\phi\boldsymbol{e}_{\mathrm{y}} + \cos\theta\boldsymbol{e}_{\mathrm{z}} \\ \boldsymbol{e}_{\theta} &= \cos\theta\cos\phi\boldsymbol{e}_{\mathrm{x}} + \cos\theta\sin\phi\boldsymbol{e}_{\mathrm{y}} - \sin\theta\boldsymbol{e}_{\mathrm{z}} \\ \boldsymbol{e}_{\phi} &= -\sin\phi\boldsymbol{e}_{\mathrm{x}} + \cos\phi\boldsymbol{e}_{\mathrm{y}}. \end{aligned}$$

Hence, for any vector valued function in two dimensions

$$\boldsymbol{\Psi} = \Psi_{\mathrm{x}}\boldsymbol{e}_{\mathrm{x}} + \Psi_{\mathrm{y}}\boldsymbol{e}_{\mathrm{y}} + \Psi_{\mathrm{z}}\boldsymbol{e}_{\mathrm{z}} = \Psi_{\mathrm{r}}\boldsymbol{e}_{\mathrm{r}} + \Psi_{\phi}\boldsymbol{e}_{\phi} + \Psi_{\theta}\boldsymbol{e}_{\theta}$$

we get the relations (by comparing each component)

$$\begin{aligned} \Psi_{\mathrm{x}} &= \Psi_{\mathrm{r}}\sin\theta\cos\phi + \Psi_{\theta}\cos\theta\cos\phi - \Psi_{\phi}\sin\phi \\ \Psi_{\mathrm{y}} &= \Psi_{\mathrm{r}}\sin\theta\sin\phi + \Psi_{\theta}\cos\theta\sin\phi - \Psi_{\phi}\cos\phi \\ \Psi_{\mathrm{z}} &= \Psi_{\mathrm{r}}\cos\theta - \Psi_{\theta}\sin\theta. \end{aligned} \tag{B.21}$$

By a corresponding argument, we have

$$\begin{aligned}\boldsymbol{e}_{\mathrm{x}} &= \sin\theta\cos\phi\boldsymbol{e}_{\mathrm{r}} + \cos\theta\cos\phi\boldsymbol{e}_{\theta} - \sin\phi\boldsymbol{e}_{\phi}\\ \boldsymbol{e}_{\mathrm{y}} &= \sin\theta\sin\phi\boldsymbol{e}_{\mathrm{r}} + \cos\theta\sin\phi\boldsymbol{e}_{\theta} + \cos\phi\boldsymbol{e}_{\phi}\\ \boldsymbol{e}_{\mathrm{z}} &= \cos\theta\boldsymbol{e}_{\mathrm{r}} - \sin\theta\boldsymbol{e}_{\theta}.\end{aligned} \tag{B.22}$$

such that

$$\begin{aligned}\Psi_{\mathrm{r}} &= \Psi_{\mathrm{x}}\sin\theta\cos\phi + \Psi_{\mathrm{y}}\sin\theta\sin\phi + \Psi_{\mathrm{z}}\cos\theta\\ \Psi_{\theta} &= \Psi_{\mathrm{x}}\cos\theta\cos\phi + \Psi_{\mathrm{y}}\cos\theta\sin\phi - \Psi_{\mathrm{z}}\sin\theta\\ \Psi_{\phi} &= -\Psi_{\mathrm{x}}\sin\phi + \Psi_{\mathrm{y}}\cos\phi.\end{aligned}$$

Using Equation (B.20), Equation (B.21) and Equation (B.22) we get

$$\nabla\psi = \frac{\partial\psi}{\partial x}\boldsymbol{e}_{\mathrm{x}} + \frac{\partial\psi}{\partial y}\boldsymbol{e}_{\mathrm{y}} + \frac{\partial\psi}{\partial z}\boldsymbol{e}_{\mathrm{z}} = \frac{\partial\psi}{\partial r}\boldsymbol{e}_{\mathrm{r}} + \frac{1}{r}\frac{\partial\psi}{\partial\theta}\boldsymbol{e}_{\theta} + \frac{1}{r\sin\theta}\frac{\partial\psi}{\partial\phi}\boldsymbol{e}_{\phi} \tag{B.23}$$

$$\Delta\psi = \frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2} + \frac{\partial^2\psi}{\partial z^2} \tag{B.24}$$

$$= \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\psi}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial\psi}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2\psi}{\partial\phi^2} \tag{B.25}$$

$$\nabla\cdot\boldsymbol{\Psi} = \frac{\partial\Psi_{\mathrm{x}}}{\partial x} + \frac{\partial\Psi_{\mathrm{y}}}{\partial y} + \frac{\partial\Psi_{\mathrm{z}}}{\partial z} = \frac{1}{r^2}\frac{\partial(r^2\Psi_{\mathrm{r}})}{\partial r} + \frac{1}{r\sin\theta}\frac{\partial(\Psi_{\theta}\sin\theta)}{\partial\theta} + \frac{1}{r\sin\theta}\frac{\partial\Psi_{\phi}}{\partial\phi} \tag{B.26}$$

$$\Delta\boldsymbol{\Psi} = \left(\Delta\Psi_{\mathrm{r}} - \frac{2}{r^2}\Psi_{\mathrm{r}} - \frac{2}{r^2\sin\theta}\frac{\partial(\Psi_{\theta}\sin\theta)}{\partial\theta} - \frac{2}{r^2\sin\theta}\frac{\partial\Psi_{\phi}}{\partial\phi}\right)\boldsymbol{e}_{\mathrm{r}} \tag{B.27}$$

$$+ \left(\Delta\Psi_{\theta} - \frac{1}{r^2\sin^2\theta}\Psi_{\theta} + \frac{2}{r^2}\frac{\partial\Psi_{\mathrm{r}}}{\partial\theta} - \frac{2\cos\theta}{r^2\sin^2\theta}\frac{\partial\Psi_{\phi}}{\partial\phi}\right)\boldsymbol{e}_{\theta} \tag{B.28}$$

$$+ \left(\Delta\Psi_{\phi} - \frac{1}{r^2\sin^2\theta}\Psi_{\phi} + \frac{2}{r^2\sin\theta}\frac{\partial\Psi_{\mathrm{r}}}{\partial\phi} + \frac{2\cos\theta}{r^2\sin^2\theta}\frac{\partial\Psi_{\theta}}{\partial\phi}\right)\boldsymbol{e}_{\phi} \tag{B.29}$$

We may also for the spherical coordinates find the relation between the stress fields as we did for the cylindrical coordinates in Appendix B.2. In spherical coordinates we get

$$[\alpha_{ij}] = \begin{bmatrix}\sin\theta\cos\phi & \sin\theta\sin\phi & \cos\theta\\ \cos\theta\cos\phi & \cos\theta\sin\phi & -\sin\theta\\ -\sin\phi & \cos\phi & 0\end{bmatrix}.$$

Which yields the relation

$$\begin{bmatrix}\sigma_{\mathrm{rr}}\\ \sigma_{\theta\theta}\\ \sigma_{\phi\phi}\\ \sigma_{\theta\phi}\\ \sigma_{\mathrm{r}\phi}\\ \sigma_{\mathrm{r}\theta}\end{bmatrix} = \boldsymbol{D}\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{33}\\ \sigma_{23}\\ \sigma_{13}\\ \sigma_{12}\end{bmatrix}$$

where

$$\boldsymbol{D} = \begin{bmatrix}\sin^2\theta\cos^2\phi & \sin^2\theta\sin^2\phi & \cos^2\theta & \sin 2\theta\sin\phi & \sin 2\theta\cos\phi & \sin^2\theta\sin 2\phi\\ \cos^2\theta\cos^2\phi & \cos^2\theta\sin^2\phi & \sin^2\theta & -\sin 2\theta\sin\phi & -\sin 2\theta\cos\phi & \cos^2\theta\sin 2\phi\\ \sin^2\phi & \cos^2\phi & 0 & 0 & 0 & -\sin 2\phi\\ -\frac{1}{2}\cos\theta\sin 2\phi & \frac{1}{2}\cos\theta\sin 2\phi & 0 & -\sin\theta\cos\phi & \sin\theta\sin\phi & \cos\theta\cos 2\phi\\ -\frac{1}{2}\sin\theta\sin 2\phi & \frac{1}{2}\sin\theta\sin 2\phi & 0 & \cos\theta\cos\phi & -\cos\theta\sin\phi & \sin\theta\cos 2\phi\\ \frac{1}{2}\sin 2\theta\cos^2\phi & \frac{1}{2}\sin 2\theta\sin^2\phi & -\frac{1}{2}\sin 2\theta & \cos 2\theta\sin\phi & \cos 2\theta\cos\phi & \frac{1}{2}\sin 2\theta\sin 2\phi\end{bmatrix}$$

The inverse relation is found by inverting the matrix $\boldsymbol{D}$.

$$\boldsymbol{D}^{-1} = \begin{bmatrix} \sin^2\theta\cos^2\phi & \cos^2\theta\cos^2\phi & \sin^2\phi & -\cos\theta\sin 2\phi & -\sin\theta\sin 2\phi & \sin 2\theta\cos^2\phi \\ \sin^2\theta\sin^2\phi & \cos^2\theta\sin^2\phi & \cos^2\phi & \cos\theta\sin 2\phi & \sin\theta\sin 2\phi & \sin 2\theta\sin^2\phi \\ \cos^2\theta & \sin^2\theta & 0 & 0 & 0 & -\sin 2\theta \\ \frac{1}{2}\sin 2\theta\sin\phi & -\frac{1}{2}\sin 2\theta\sin\phi & 0 & -\sin\theta\cos\phi & \cos\theta\cos\phi & \sin\phi\cos 2\theta \\ -\sin\theta\sin 2\phi & -\frac{1}{2}\sin 2\theta\cos\phi & 0 & \sin\theta\sin\phi & -\cos\theta\sin\phi & \cos 2\theta\cos\phi \\ \frac{1}{2}\sin^2\theta\sin 2\phi & \frac{1}{2}\cos^2\theta\sin 2\phi & -\frac{1}{2}\sin 2\phi & \cos\theta\cos 2\phi & \cos 2\phi\sin\theta & \frac{1}{2}\sin 2\theta\sin 2\phi \end{bmatrix}$$

Moreover, we have

$$\begin{bmatrix} \sigma_{\mathrm{rr}} \\ \sigma_{\theta\theta} \\ \sigma_{\phi\phi} \\ \sigma_{\theta\phi} \\ \sigma_{\mathrm{r}\phi} \\ \sigma_{\mathrm{r}\theta} \end{bmatrix} = \boldsymbol{C} \begin{bmatrix} \varepsilon_{\mathrm{rr}} \\ \varepsilon_{\theta\theta} \\ \varepsilon_{\phi\phi} \\ 2\varepsilon_{\theta\phi} \\ 2\varepsilon_{r\phi} \\ 2\varepsilon_{\mathrm{r}\theta} \end{bmatrix}, \tag{B.30}$$

where (see [45] for details)

$$\begin{aligned} \varepsilon_{\mathrm{rr}} &= \frac{\partial u_{\mathrm{r}}}{\partial r} \\ \varepsilon_{\theta\theta} &= \frac{1}{r}\left(\frac{\partial u_{\theta}}{\partial\theta} + u_{\mathrm{r}}\right) \\ \varepsilon_{\phi\phi} &= \frac{1}{r\sin\theta}\left(\frac{\partial u_{\phi}}{\partial\phi} + u_{\mathrm{r}}\sin\theta + u_{\theta}\cos\theta\right) \\ \varepsilon_{\theta\phi} &= \frac{1}{2r}\left(\frac{1}{\sin\theta}\frac{\partial u_{\theta}}{\partial\phi} + \frac{\partial u_{\phi}}{\partial\theta} - u_{\phi}\cot\theta\right) \\ \varepsilon_{\mathrm{r}\phi} &= \frac{1}{2}\left(\frac{1}{r\sin\theta}\frac{\partial u_{\mathrm{r}}}{\partial\phi} + \frac{\partial u_{\phi}}{\partial r} - \frac{u_{\phi}}{r}\right) \\ \varepsilon_{\mathrm{r}\theta} &= \frac{1}{2}\left(\frac{1}{r}\frac{\partial u_{\mathrm{r}}}{\partial\theta} + \frac{\partial u_{\theta}}{\partial r} - \frac{u_{\theta}}{r}\right). \end{aligned} \tag{B.31}$$

B.4 The prolate spheroidal coordinate system

The prolate spheroidal coordinate system is an extension of the spherical coordinate system. It is defined by the relations

$$\begin{aligned} x &= \sqrt{r^2 - f^2}\sin\theta\cos\phi \\ y &= \sqrt{r^2 - f^2}\sin\theta\sin\phi \\ z &= r\cos\theta \end{aligned}$$

with foci located at $z = \pm f$. Note that we define $r \geq f$ such that the coordinate system reduces to the spherical coordinate system when $f = 0$. Using this condition, we may

establish the following inverse formulas

$$\begin{aligned} r &= \frac{1}{\sqrt{2}}\sqrt{T_f + \sqrt{T_f^2 - 4f^2z^2}} \\ \theta &= \arccos\left(\frac{\sqrt{2}z}{\sqrt{T_f + \sqrt{T_f^2 - 4f^2z^2}}}\right) \\ \phi &= \operatorname{atan2}(y, x). \end{aligned} \tag{B.32}$$

where $T_f = T_f(x, y, z) = x^2 + y^2 + z^2 + f^2$.

The derivatives are found to be

$$\begin{aligned} \frac{\partial x}{\partial r} &= \frac{r\sin\theta\cos\phi}{\sqrt{r^2 - f^2}}, & \frac{\partial y}{\partial r} &= \frac{r\sin\theta\sin\phi}{\sqrt{r^2 - f^2}}, & \frac{\partial z}{\partial r} &= \cos\theta \\ \frac{\partial x}{\partial \theta} &= \sqrt{r^2 - f^2}\cos\theta\cos\phi, & \frac{\partial y}{\partial \theta} &= \sqrt{r^2 - f^2}\cos\theta\sin\phi, & \frac{\partial z}{\partial \theta} &= -r\sin\theta \\ \frac{\partial x}{\partial \phi} &= -\sqrt{r^2 - f^2}\sin\theta\sin\phi, & \frac{\partial y}{\partial \phi} &= \sqrt{r^2 - f^2}\sin\theta\cos\phi, & \frac{\partial z}{\partial \phi} &= 0. \end{aligned} \tag{B.33}$$

and

$$\begin{aligned} \frac{\partial r}{\partial x} &= \frac{xr}{2r^2 - T_f}, & \frac{\partial r}{\partial y} &= \frac{yr}{2r^2 - T_f}, & \frac{\partial r}{\partial z} &= \frac{z}{r}\frac{r^2 - f^2}{2r^2 - T_f} \\ \frac{\partial \theta}{\partial x} &= \frac{xz}{r\sin\theta(2r^2 - T_f)}, & \frac{\partial \theta}{\partial y} &= \frac{yz}{r\sin\theta(2r^2 - T_f)}, & \frac{\partial \theta}{\partial z} &= \frac{1}{r\sin\theta}\left(\frac{z^2}{r^2}\frac{r^2 - f^2}{2r^2 - T_f} - 1\right) \\ \frac{\partial \phi}{\partial x} &= -\frac{y}{x^2 + y^2}, & \frac{\partial \phi}{\partial y} &= \frac{x}{x^2 + y^2}, & \frac{\partial \phi}{\partial z} &= 0. \end{aligned} \tag{B.34}$$

Using Equation (B.1) we have

$$\begin{aligned} h_\mathrm{r} &= \sqrt{\frac{r^2 - f^2\cos^2\theta}{r^2 - f^2}} \\ h_\theta &= \sqrt{r^2 - f^2\cos^2\theta} \\ h_\phi &= \sqrt{r^2 - f^2}\sin\theta \end{aligned}$$

As the prolate spheroidal coordinate system is an orthogonal coordinate system, the general nabla operator developed in Appendix B.1 reduces to

$$\nabla = \frac{\boldsymbol{e}_\mathrm{r}}{h_\mathrm{r}}\frac{\partial}{\partial r} + \frac{\boldsymbol{e}_\theta}{h_\theta}\frac{\partial}{\partial \theta} + \frac{\boldsymbol{e}_\phi}{h_\phi}\frac{\partial}{\partial \phi}$$

The determinant of the jacobian in Equation (4.20) may now be written as

$$J_1 = h_\mathrm{r}h_\theta h_\phi = \left(r^2 - f^2\cos^2\theta\right)\sin\theta$$

The surface Jacobian at a given (constant) $r = \hat{r}$ is

$$J_S = h_\theta h_\phi = \sqrt{r^2 - f^2\cos^2\theta}\sqrt{r^2 - f^2}\sin\theta.$$

Note that

$$\lim_{\hat{r}\to\infty} J_S = \hat{r}^2 \sin\theta.$$

That is, in the limit $\hat{r} \to \infty$ the surface Jacobian of the prolate spheroidal coordinate system is the same as the surface Jacobian of the spherical coordinate system.

Appendix C

Analytic solutions and convergence analysis

When a finite element method program is developed it is crucial to get numerical evidence through analytic solution and benchmark solutions. In this appendix we develop some analytic solution which will be used to verify the correctness of the implementations. We have also tested the program on some well known benchmark solutions. Some results of these studies have been moved to this appendix to reduce the size of the main text.

C.1 Shell obstacle course

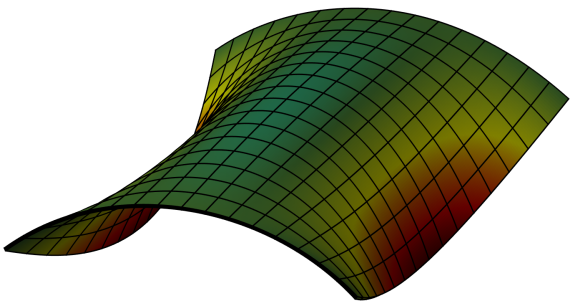
Exact solution in terms of shell analysis have been presented in [46]. For a more detailed problem description than presented here, we refer to [2] or [46].

C.1.1 Scordelis-Lo Roof

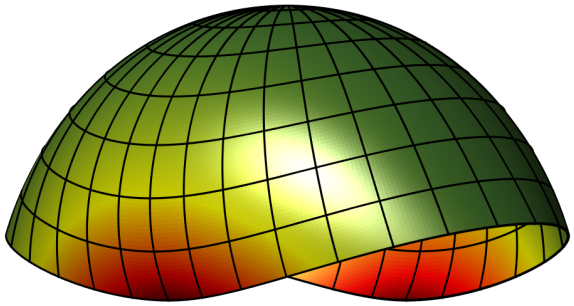
Mesh 1, 2, 3 and 6 are shown in Figure C.2. The data for the coarsest mesh (mesh 1) can be found in Appendix D.3 with $\phi = 40°$ and $\theta = \frac{\pi}{2} - \phi$. Note that due to symmetry, only one quadrant is analyzed. In Figure C.3 we plot the convergence of the displacement (absolute value) at the mid span of the side edge. A visualization of the result is presented in Figure C.1a.

C.1.2 Pinched hemisphere

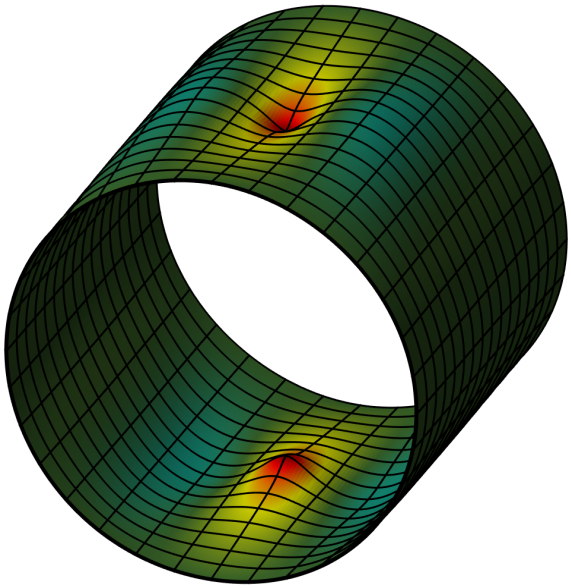
A thin hemisphere is pinched at antipodal points of the equator with equal but opposite directed forces. The equator is else considered to be free, and the north pole is fixed. Mesh 1, 2, 3 and 6 are shown in Figure C.4. Note that due to symmetry, only one quadrant is analyzed. In Figure C.5 we plot the convergence of the displacement (absolute value) at the point load. The plot is almost identical to the one presented by Hughes et al. in [2], the largest difference is the quartic case where we have established a better approximation



(a) Scordelis-Lo Roof.

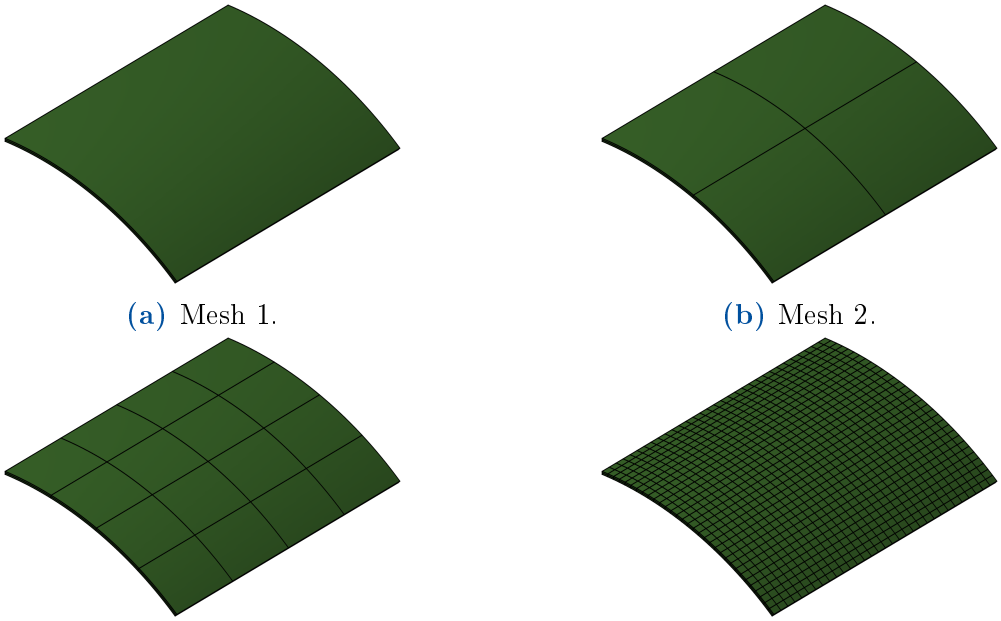


(b) Pinched hemisphere.



(c) Pinched cylinder.

Figure C.1: Shell obstacle course: Displacement plots. For Scordelis-Lo roof (a), the countours are based on the displacement in the direction of the gravity loading (scaling factor 20 is used). For the pinched hemisphere (b), the contours are based on the displacement in the inward directed point load (scaling factor 33.3 is used). For the pinched cylinder (c), the contours are based on the displacement in the direction of the point load (scaling factor $3 \cdot 10^6$ is used). The polynomial order $p = q = 5$ and mesh 4 was used in all cases (where convergence is obtained).



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 6.

Figure C.2: Scordeli-Lo Roof: Meshes. If s is the mesh number, then the number of inserted knots in ξ- and η-direction is given by $2^{s-1}-1$. Thus, the number of surface control points per side (in ξ- and η-direction) is given by $n = m = 2^{s-1} + p = 2^{s-1} + q$.

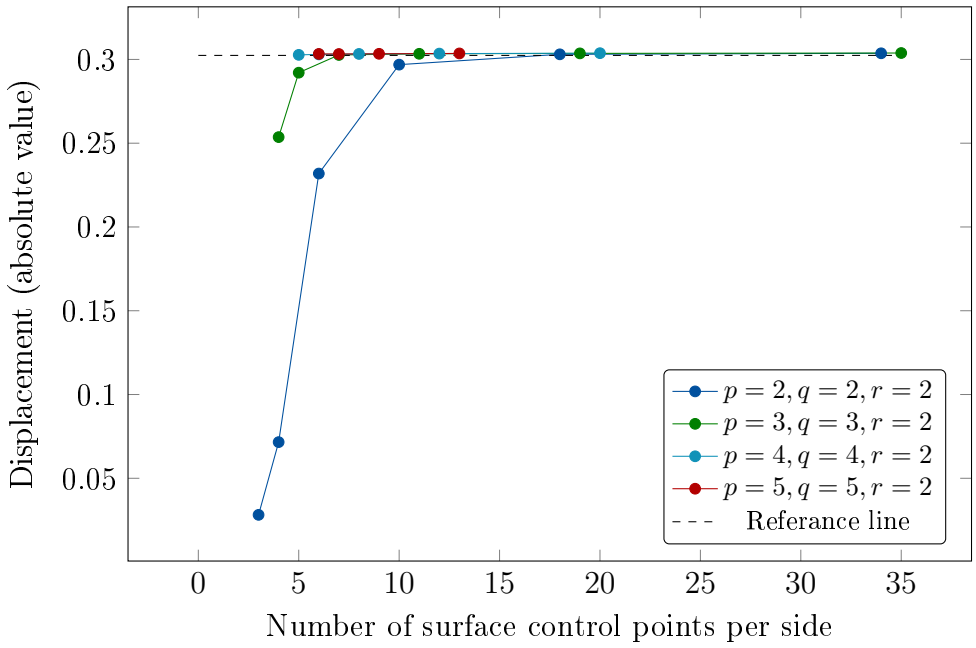
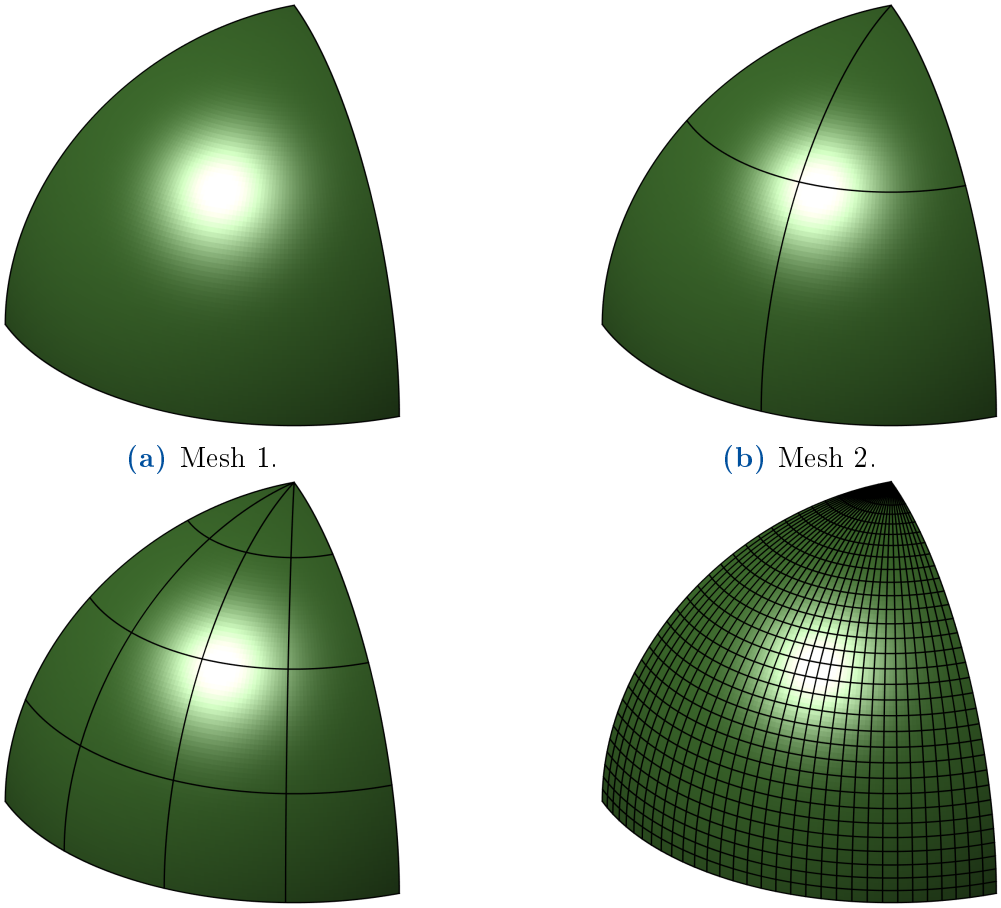


Figure C.3: Scordeli-Lo Roof: Convergence plot.



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 6.

Figure C.4: Pinched hemisphere: Meshes. If s is the mesh number, then the number of inserted knots in ξ- and η-direction is given by $2^{s-1}-1$. Thus, the number of surface control points per side (in ξ- and η-direction) is given by $n = m = 2^{s-1} + p = 2^{s-1} + q$.

on mesh 1[1]. In Figure C.6 we plot the convergence (of the same displacement) with only one NURBS element when $p = q$ is elevated. After 10 elevations we basically have achieved the desired result, which shows the elegance of the isogeometric analysis as it easily allows us to elevate the order without changing the geometry. Once again, we have obtained the same results as in [2], except for $p = q = 4$. A visualization of the result is presented in Figure C.1b.

C.1.3 Pinched cylinder

A thin cylindrical shell is pinched at its midspan with a corresponding equal but opposite directed force on the opposite side of the cylinder. The two ends of the cylinder are supported by rigid diaphragms. Mesh 1, 2, 3 and 7 are shown in Figure C.7. Note that due to symmetry, only one octant is analyzed. In Figure C.8 we plot the convergence of the displacement (absolute value) at the point load. The plot is once again almost identical to the one presented by Hughes et al. in [2]. The number of control points per

[1] As this is the only point which is notably different from our results, this may simply be a printing error in [2].

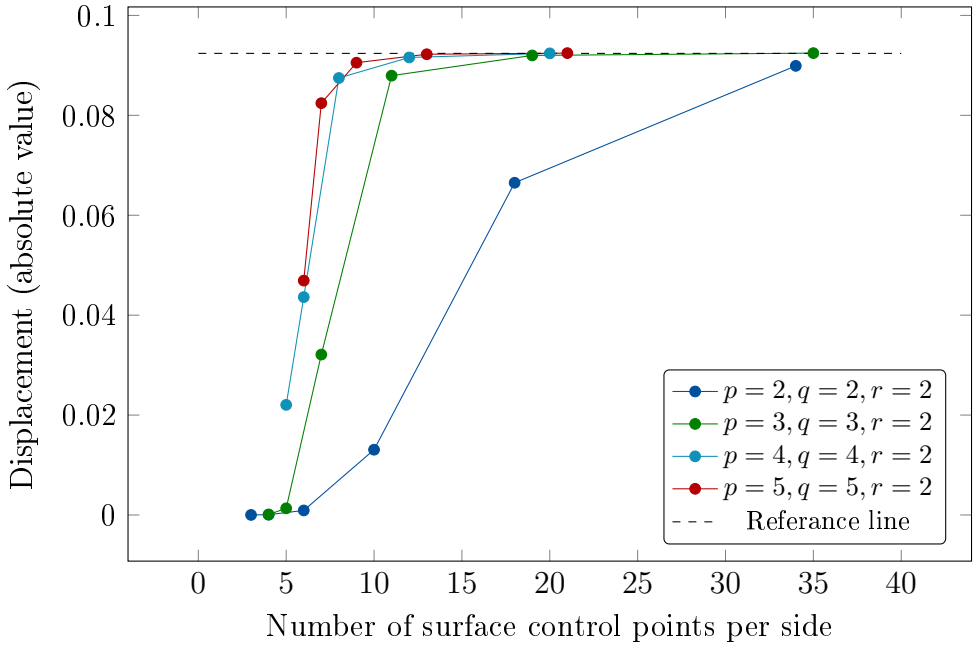


Figure C.5: Pinched hemisphere: Convergence plot.

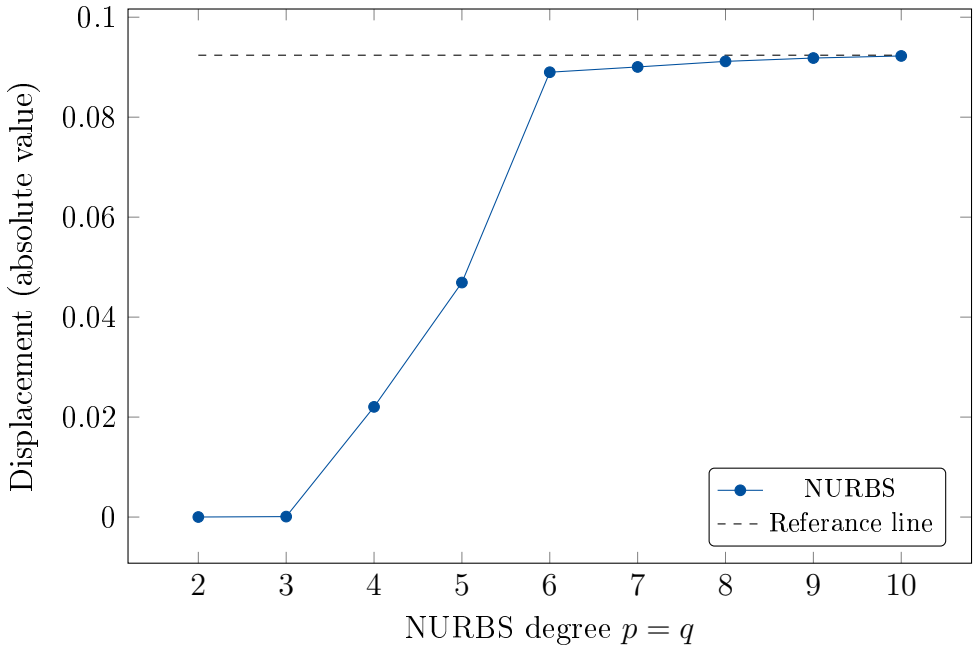
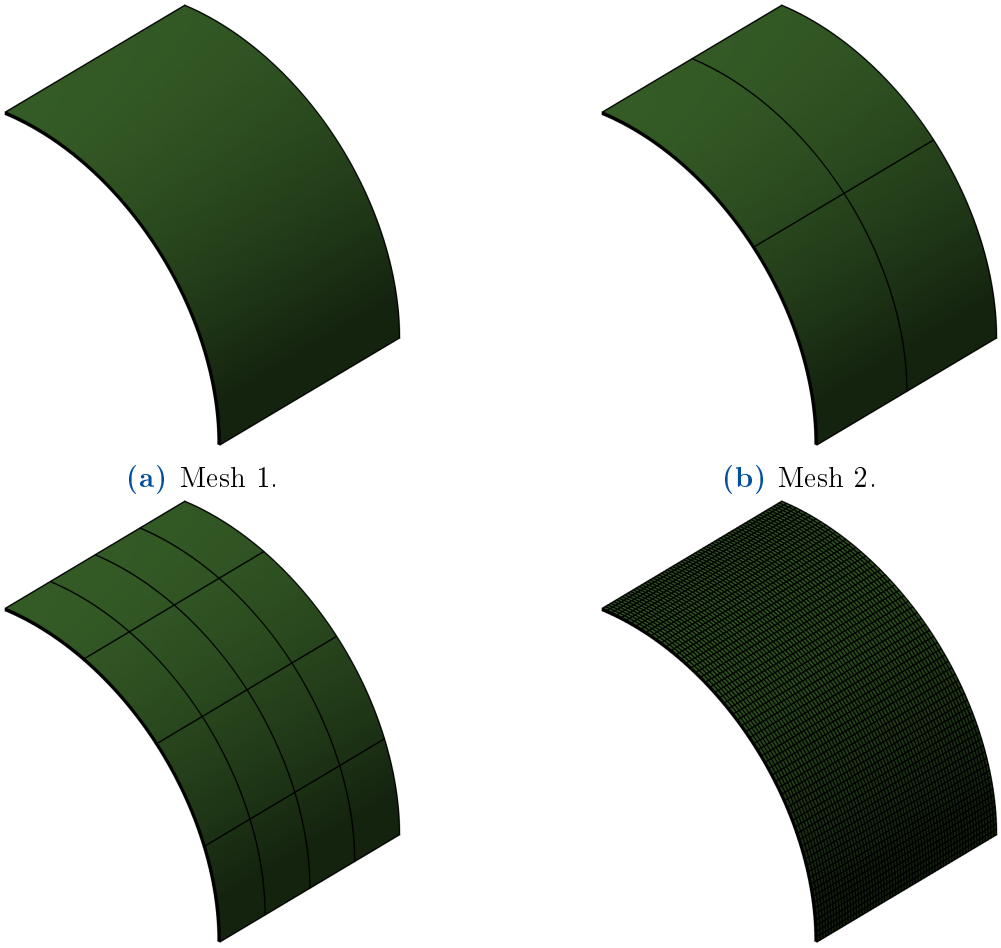


Figure C.6: Pinched hemisphere: Convergence plot using only one NURBS element (note that $r = 2$ is constant in this analysis such that we have $l = 3$).



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 7.

Figure C.7: Pinched cylinder: Meshes. If s is the mesh number, then the number of inserted knots in ξ- and η-direction is given by $2^{s-1} - 1$. Thus, the number of surface control points per side (in ξ- and η-direction) is given by $n = m = 2^{s-1} + p = 2^{s-1} + q$.

side should be different for the cubic and quartic case for mesh 6 (respectively $n = 35$ and $n = 36$) so this must be a print error in [2].

A visualization of the result is presented in Figure C.1c.

C.2 Helmholtz equation

We shall now solve Helmholtz equation given by

$$\Delta u + k^2 u = 0$$

in spherical coordinates assuming the solution to have the form $u = f(r)g(\theta)h(\phi)$. The Laplace operator in spherical coordinates takes the form

$$\Delta u(r, \theta, \phi) = \frac{1}{r^2}\left[\frac{\partial}{\partial r}\left(r^2 \frac{\partial u}{\partial r}\right) + \frac{1}{\sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta \frac{\partial u}{\partial \theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial \phi^2}\right]$$

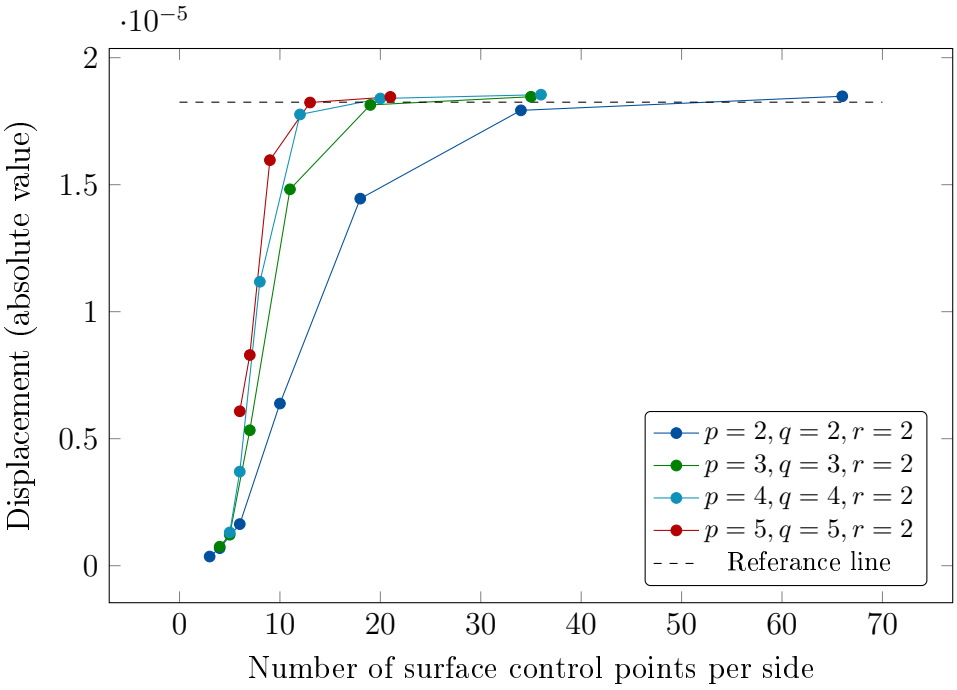


Figure C.8: Pinched cylinder: Convergence plot.

Thus, inserting $u = f(r)g(\theta)h(\phi)$ into

$$\frac{1}{r^2}\left[\frac{\partial}{\partial r}\left(r^2\frac{\partial u}{\partial r}\right) + \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial u}{\partial\theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right] + k^2 u = 0$$

and dividing the resulting equation by $f(r)g(\theta)h(\phi)$ we get

$$\frac{1}{r^2}\left[\frac{1}{f}\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2\frac{\mathrm{d}f}{\mathrm{d}r}\right) + \frac{1}{g}\frac{1}{\sin\theta}\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}g}{\mathrm{d}\theta}\right) + \frac{1}{h}\frac{1}{\sin^2\theta}\frac{\mathrm{d}^2 h}{\mathrm{d}\phi^2}\right] + k^2 = 0$$

or

$$\left[\frac{1}{f}\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2\frac{\mathrm{d}f}{\mathrm{d}r}\right) + k^2 r^2\right]\sin^2\theta + \frac{1}{g}\sin\theta\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}g}{\mathrm{d}\theta}\right) = -\frac{1}{h}\frac{\mathrm{d}^2 h}{\mathrm{d}\phi^2}.$$

As the left hand side only depends on r and θ and the right hand side only depends on ϕ, each side must be equal to a separation constant, call it γ_1. The equation is thus *separated* into the equations

$$\frac{\mathrm{d}^2 h}{\mathrm{d}\phi^2} + \gamma_1 h = 0$$

and

$$\frac{1}{f}\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2\frac{\mathrm{d}f}{\mathrm{d}r}\right) + k^2 r^2 = \frac{1}{\sin^2\theta}\left[\gamma_1 - \sin\theta\frac{1}{g}\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}g}{\mathrm{d}\theta}\right)\right].$$

The latter equation may once again, using the same argument, be separated in two new equations by a separation constant γ_2

$$\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2\frac{\mathrm{d}f}{\mathrm{d}r}\right) + (k^2 r^2 - \gamma_2)f = 0$$

and

$$\sin\theta\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}g}{\mathrm{d}\theta}\right) + (\gamma_2\sin^2\theta - \gamma_1)g = 0.$$

We have now reduced the problem to three separate ordinary differential equations

$$\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2\frac{\mathrm{d}f(r)}{\mathrm{d}r}\right)+(k^2r^2-\gamma_2)f(r)=0 \tag{C.1}$$

$$\sin\theta\frac{\mathrm{d}}{\mathrm{d}\theta}\left(\sin\theta\frac{\mathrm{d}g(\theta)}{\mathrm{d}\theta}\right)+(\gamma_2\sin^2\theta-\gamma_1)g(\theta)=0 \tag{C.2}$$

$$\frac{\mathrm{d}^2h(\phi)}{\mathrm{d}\phi^2}+\gamma_1h(\phi)=0. \tag{C.3}$$

As we only consider domains enclosing the origin, we must have the periodic condition $h(0)=h(2\pi)$. If $\gamma_1=0$, h is just the constant function (as the linear term must vanish due to the periodic condition). In all other cases it must be of the form

$$h(\phi)=C_1\mathrm{e}^{-r\phi}+C_2\mathrm{e}^{r\phi}$$

where $r=\sqrt{-\gamma_1}$ could be a complex constant. The harmonic time dependency assumption her implies that $\gamma_1=m^2$, for some $m\in\mathbb{Z}\setminus\{0\}$. Due to symmetry, we can redefine constants C_1 and C_2 such that m is only positive. Hence, if $m=0$ ($\gamma_1=0$) then h is the constant function, and if $m\in\mathbb{N}$ then

$$h(\phi)=C_1\mathrm{e}^{-\mathrm{i}m\phi}+C_2\mathrm{e}^{\mathrm{i}m\phi}.$$

Consider now equation Equation (C.2) with $\gamma_1=m^2$. We use the transformation $w=\cos\theta$ (such that $1-w^2=\sin^2\theta$) to obtain

$$(1-w^2)\frac{\mathrm{d}^2g(w)}{\mathrm{d}w^2}-2t\frac{\mathrm{d}g(w)}{\mathrm{d}w}+\left(\gamma_2-\frac{m^2}{1-w^2}\right)g(w)=0.$$

for $\gamma_2=n(n+1)$ where $n\in\mathbb{N}_0$, this is the Legendre's equation[2] which have the associated Legendre functions

$$g_{mn}(\theta)=P_n^m(\cos\theta)$$

as solutions. These functions are defined from the Legendre polynomials $P_n(t)$ by

$$P_n^m(w)=(1-w^2)^{m/2}\frac{\mathrm{d}^mP_n(t)}{\mathrm{d}w^m}\quad 0\le m\le n.$$

Finally if $\gamma_2=n(n+1)$, Equation (C.1) is the spherical Bessel equation with the spherical Bessel functions as solutions

$$f(r)=A_nj_n(kr)+B_ny_n(kr).$$

The full solution of Helmholtz equation is then, by superposition, given by

$$\begin{aligned}u(r,\theta,\phi)&=\sum_{n=0}^{\infty}(A_nj_n(kr)+B_ny_n(kr))\sum_{m=0}^{n}P_n^m(\cos\theta)(C_{nm}\mathrm{e}^{-\mathrm{i}m\phi}+D_{nm}\mathrm{e}^{\mathrm{i}m\phi})\\&=\sum_{n=0}^{\infty}(A_nj_n(kr)+B_ny_n(kr))\sum_{m=-n}^{n}P_n^m(\cos\theta)C_{nm}^*\mathrm{e}^{-\mathrm{i}m\phi}\end{aligned}$$

Recalling the definition of the spherical Hankel functions of first and second kind ($h_n^{(1)}(x)$ and $h_n^{(2)}(x)$, respectively) we may write

$$u(r,\theta,\phi)=\sum_{n=0}^{\infty}(A_n^*h_n^{(1)}(kr)+B_n^*h_n^{(2)}(kr))\sum_{m=-n}^{n}P_n^m(\cos\theta)C_{nm}^*\mathrm{e}^{-\mathrm{i}m\phi}.$$

[2] The nontrivial question why γ_2 must take these values will not be answered here. The reader is advised to turn to more advanced literature.

C.3 Analytic solutions of elasticity problem

In this section we shall present the first steps in solving the scattering problem on a spherical shell. For the full treatment of the problem, cf. [14].

We start by writing Equation (2.15) in vector form. We thus need to rewrite $\sigma_{ij,j}$ in therms of the vector solution $\boldsymbol{u}$. Recall that

$$\sigma_{ij} = c_{ijkl}\varepsilon_{kl} = \left[\lambda\delta_{ij}\delta_{kl} + \mu\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right)\right]\varepsilon_{kl}$$

where

$$\varepsilon_{ij} = \frac{u_{i,j} + u_{j,i}}{2}.$$

Hence,

$$\begin{aligned}
\begin{bmatrix}\sigma_{1j,j}\\ \sigma_{2j,j}\\ \sigma_{3j,j}\end{bmatrix} &= \sum_{j=1}^{3}\begin{bmatrix}\sigma_{1j,j}\\ \sigma_{2j,j}\\ \sigma_{3j,j}\end{bmatrix} = \sum_{j=1}^{3}\sum_{k=1}^{3}\sum_{l=1}^{3}\frac{1}{2}\begin{bmatrix}\left[\lambda\delta_{1j}\delta_{kl} + \mu\left(\delta_{1k}\delta_{jl} + \delta_{1l}\delta_{jk}\right)\right]\left(u_{k,jl} + u_{l,jk}\right)\\ \left[\lambda\delta_{2j}\delta_{kl} + \mu\left(\delta_{2k}\delta_{jl} + \delta_{2l}\delta_{jk}\right)\right]\left(u_{k,jl} + u_{l,jk}\right)\\ \left[\lambda\delta_{3j}\delta_{kl} + \mu\left(\delta_{3k}\delta_{jl} + \delta_{3l}\delta_{jk}\right)\right]\left(u_{k,jl} + u_{l,jk}\right)\end{bmatrix}\\
&= \begin{bmatrix}\mu(2u_{1,11} + u_{2,21} + u_{1,22} + u_{3,31} + u_{1,33}) + \lambda(u_{1,11} + u_{2,12} + u_{3,13})\\ \mu(u_{2,11} + u_{1,12} + 2u_{2,22} + u_{3,32} + u_{2,33}) + \lambda(u_{1,21} + u_{2,22} + u_{3,23})\\ \mu(u_{3,11} + u_{1,13} + u_{3,22} + u_{2,23} + 2u_{3,33}) + \lambda(u_{1,31} + u_{2,32} + u_{3,33})\end{bmatrix}\\
&= \begin{bmatrix}\mu(u_{1,11} + u_{1,22} + u_{1,33}) + (\mu+\lambda)(u_{1,11} + u_{2,12} + u_{3,13})\\ \mu(u_{2,11} + u_{2,22} + u_{2,33}) + (\mu+\lambda)(u_{1,21} + u_{2,12} + u_{3,23})\\ \mu(u_{3,11} + u_{3,22} + u_{3,33}) + (\mu+\lambda)(u_{1,31} + u_{2,12} + u_{3,33})\end{bmatrix}
\end{aligned}$$

where we in the finale step have assumed the mixed partial derivatives to be equal (that is $u_{i,kl} = u_{i,lk} \quad \forall i,l,k = 1,2,3$).

Note that the i^{th} component of $\Delta\boldsymbol{u} = \{\Delta u_1, \Delta u_2, \Delta u_3\}^\top$ is given by

$$(\Delta\boldsymbol{u})_i = u_{i,11} + u_{i,22} + u_{i,33}$$

and the i^{th} component of $\nabla(\nabla\cdot\boldsymbol{u})$ is given by

$$[\nabla(\nabla\cdot\boldsymbol{u})]_i = \nabla(u_{1,1} + u_{2,2} + u_{3,3}) = u_{1,i1} + u_{2,i2} + u_{3,i3}.$$

Thus,

$$\mu\Delta\boldsymbol{u} + (\lambda+\mu)\nabla(\nabla\cdot\boldsymbol{u}) + \rho_{\text{s}}\omega^2\boldsymbol{u} = \boldsymbol{0}. \tag{C.4}$$

From here, we assume the solution to take the form

$$\boldsymbol{u} = \nabla\Phi + \nabla\times\boldsymbol{\Psi} \tag{C.5}$$

which inserted in Equation (C.4) yields

$$\mu\Delta(\nabla\Phi + \nabla\times\boldsymbol{\Psi}) + (\lambda+\mu)\nabla(\nabla\cdot(\nabla\Phi + \nabla\times\boldsymbol{\Psi})) + \rho_{\text{s}}\omega^2(\nabla\Phi + \nabla\times\boldsymbol{\Psi}) = \boldsymbol{0}.$$

Using the relations

$$\begin{gathered}
\nabla\cdot\nabla\Phi = \Delta\Phi, \quad \nabla\times(\nabla\Phi) = \boldsymbol{0}, \quad \nabla\cdot(\nabla\times\boldsymbol{\Psi}) = 0,\\
\Delta(\nabla\Phi) = \nabla(\Delta\Phi), \quad \Delta(\nabla\times\Psi) = \nabla\times(\Delta\Psi)
\end{gathered}$$

we get

$$\mu\Delta(\nabla\Phi + \nabla\times\boldsymbol{\Psi}) + (\lambda+\mu)\nabla(\Delta\Phi + \mathbf{0}) + \rho_s\omega^2(\nabla\Phi + \nabla\times\boldsymbol{\Psi}) = \mathbf{0}.$$
$$\Updownarrow$$
$$\nabla(\mu\Delta\Phi) + \nabla\times(\mu\Delta\Psi) + (\lambda+\mu)\nabla(\Delta\Phi) + \rho_s\omega^2\nabla\Phi + \rho_s\omega^2\nabla\times\boldsymbol{\Psi} = \mathbf{0}.$$
$$\Updownarrow$$
$$\nabla((2\mu+\lambda)\Delta\Phi + \rho_s\omega^2\Phi) + \nabla\times(\mu\Delta\boldsymbol{\Psi} + \rho_s\omega^2\boldsymbol{\Psi}) = \mathbf{0}.$$

Defining the longitudinal wave velocity and shear wave velocity by

$$c_1 = \sqrt{\frac{\lambda+2\mu}{\rho_s}} \quad \text{and} \quad c_2 = \sqrt{\frac{\mu}{\rho_s}}, \tag{C.6}$$

respectively, we arrive at

$$\nabla(c_1^2\Delta\Phi + \omega^2\Phi) + \nabla\times(c_2^2\Delta\boldsymbol{\Psi} + \omega^2\boldsymbol{\Psi}) = \mathbf{0}.$$

which is satisfied if

$$\Delta\Phi + \alpha^2\Phi = 0 \quad \text{and} \quad \Delta\boldsymbol{\Psi} + \beta^2\boldsymbol{\Psi} = \mathbf{0}$$

where

$$\alpha = \frac{\omega}{c_1} \quad \text{and} \quad \beta = \frac{\omega}{c_2}.$$

Hence, we have arrived at two Helmholtz equation where the second is in vector form. Obviously, the vector form result in three Helmholtz equation in scalar form (one for each of the three components of $\boldsymbol{\Psi}$). Hence, if we have the solution to the Helmholtz, we have found the scalar potential Φ and the vector potential Ψ. The solution of Equation (C.4) is then computed by Equation (C.5).

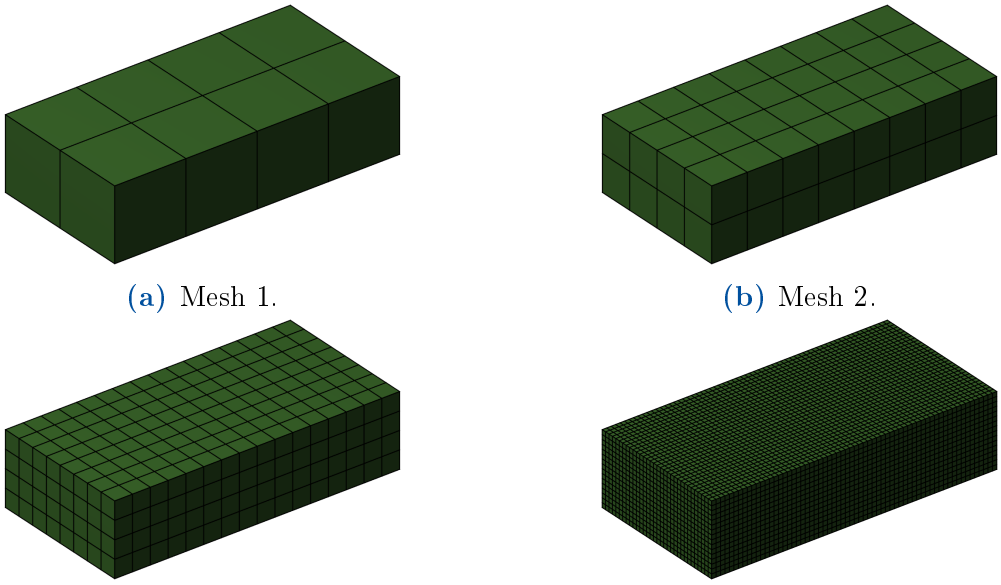
C.4 Elasticity problems in Cartesian coordinates

We start by exploring analytic solutions on the simplest domain, namely a rectangular prism. Its width in x-, y-, and z-direction is w_x, w_y and w_z, respectively. That is,

$$\Omega = (0, w_x)\times(0, w_y)\times(0, w_z)$$

On this domain, we shall present four analytic solution, where we let $w_x = 4$, $w_y = 2$ and $w_z = 1$. Each solution will provide evidence for different aspects of a finite element method program. In all test cases we shall construct solution which have homogeneous Dirichlet boundary conditions at $x = 0$. In the first case we have non homogeneous Neumann conditions on the remaining sides, while the remaining three cases have homogeneous Dirichlet boundary conditions on the whole boundary. Given a solution, we use Maple to find the corresponding boundary conditions and body loadings.

Some meshes are shown in Figure C.9. Note that we insert the knots $\{0.25, 0.5, 0.75\}$ in ξ-direction (x-direction) and the knot 0.5 in η direction (y-direction). This is done to get an ideally *aspect ratio*, which is the measure of a mesh element's deviation from having all sides of equal length. One can observe that one might get a slow start in the convergence if the aspect ratio is very high.



(a) Mesh 1. **(b)** Mesh 2.

(c) Mesh 3. **(d)** Mesh 5.

Figure C.9: Rectangular prism: Meshes.

The first case (named Rectangular prism 1) is given by

$$\boldsymbol{u} = \begin{bmatrix} \frac{1}{5}xy(2\nu^2x^2 - 6\nu^2z^2 - 4\nu x^2 + 15\nu z^2 - 6z^2) \\ \frac{1}{10}x^2(\nu x^2 - 3\nu z^2 - 2x^2 + 6z^2) \\ -\frac{3}{5}x^2yz(\nu - 2) \end{bmatrix}. \tag{C.7}$$

The analysis is done with first, second and third order NURBS, so since the analytic solution has a fourth order polynomial in the y direction, the solution is not in the search space $\boldsymbol{\mathcal{S}}^h$. The convergence plot is presented in Figure C.10.

Consider now the solution (rectangular prism 2)

$$\boldsymbol{u} = \begin{bmatrix} x(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \\ x(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \\ x(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \end{bmatrix}. \tag{C.8}$$

It does indeed satisfy homogeneous Dirichlet conditions with a rather ugly resulting function $\boldsymbol{f}$. As this solution only has degree two for each polynomial in each of the spatial direction, the solution is an element in the search space $\boldsymbol{\mathcal{S}}^h$ if second or higher order NURBS is used. We should thus expect the numerical solution to be exact (to machine precision) for these cases. This is indeed the case as well. The convergence for first order NURBS are illustrated in Figure C.11.

By multiplying the solution in (C.8) by another factor x,

$$\boldsymbol{u} = \begin{bmatrix} x^2(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \\ x^2(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \\ x^2(x - w_{\mathrm{x}})y(y - w_{\mathrm{y}})z(z - w_{\mathrm{z}}) \end{bmatrix}, \tag{C.9}$$

we get a solution which should not be in the search space $\boldsymbol{\mathcal{S}}^h$ unless we use third order (or higher) NURBS, and this is indeed the case. The homogeneous Dirichlet boundary

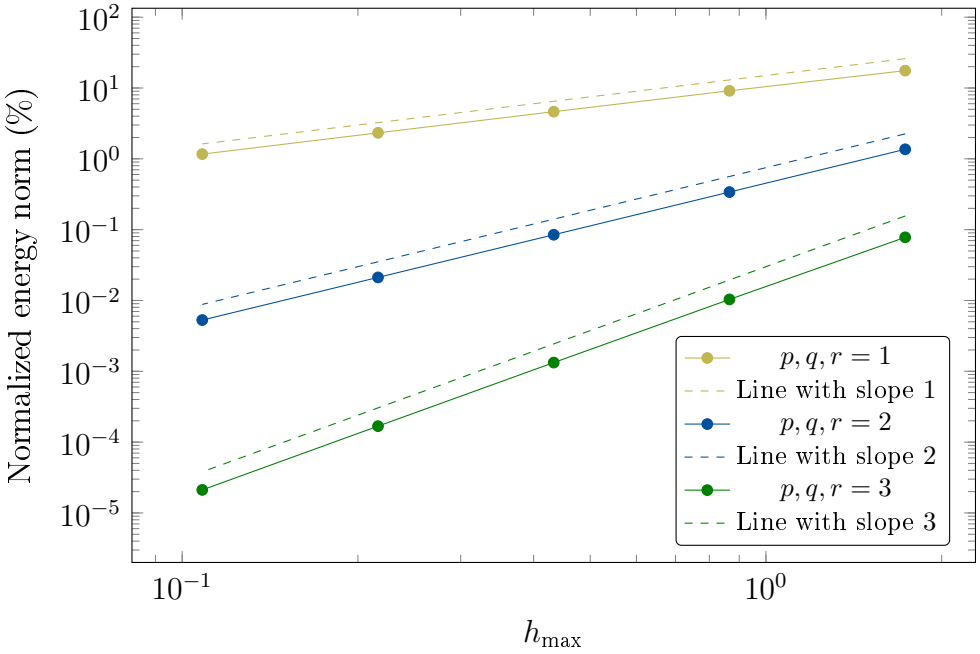


Figure C.10: Rectangular prism 1: Convergence plot.

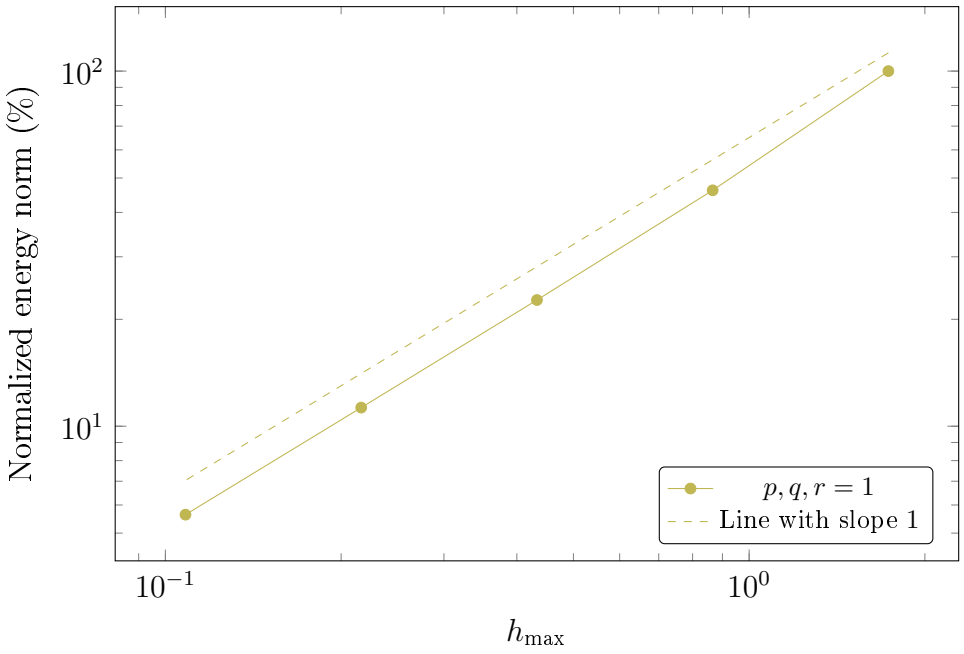


Figure C.11: Rectangular prism 2: Convergence plot.

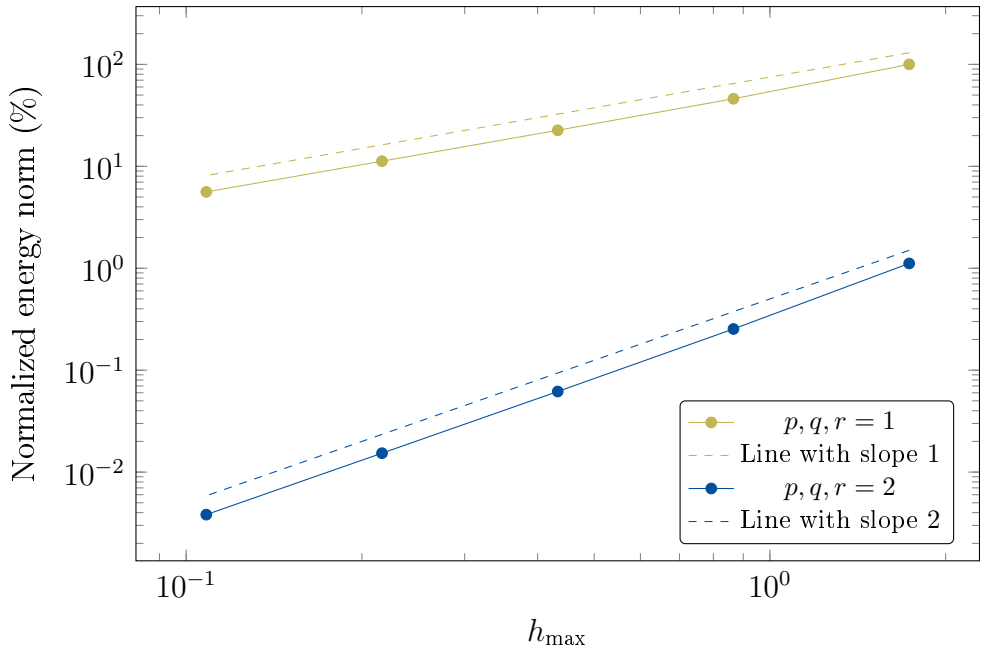


Figure C.12: Rectangular prism 3: Convergence plot.

condition of course still hold, but now we have an even more complex function $\boldsymbol{f}$. The convergence plot is given in Figure C.12.

All of the discussed solutions have been polynomial solution, and would thus potentially be in the solution space $\boldsymbol{\mathcal{S}}^h$ if we elevate the order of the NURBS. Thus, the solutions presented so far does not fit so much for analysis on the higher order elevation of NURBS. A trigonometric solution given by

$$\boldsymbol{u} = \begin{bmatrix} \sin\left(\frac{4\pi x}{w_{\mathrm{x}}}\right) \sin\left(\frac{3\pi y}{w_{\mathrm{y}}}\right) \sin\left(\frac{2\pi z}{w_{\mathrm{z}}}\right) \\ \sin\left(\frac{4\pi x}{w_{\mathrm{x}}}\right) \sin\left(\frac{3\pi y}{w_{\mathrm{y}}}\right) \sin\left(\frac{2\pi z}{w_{\mathrm{z}}}\right) \\ \sin\left(\frac{4\pi x}{w_{\mathrm{x}}}\right) \sin\left(\frac{3\pi y}{w_{\mathrm{y}}}\right) \sin\left(\frac{2\pi z}{w_{\mathrm{z}}}\right) \end{bmatrix}, \tag{C.10}$$

will also satisfy homogeneous Dirichlet boundary conditions at the boundary. The convergence plot is given in Figure C.13. It does now not matter how many times we order elevate as the exact solution will never lie in the search space.

C.5 Elasticity problem in cylindrical coordinates

We shall here develop an analytic solution to the static elasticity problem given by

$$\mu \Delta \boldsymbol{u} + (\lambda + \mu) \nabla (\nabla \cdot \boldsymbol{u}) = \boldsymbol{0} \tag{C.11}$$

(cf. Equation (C.4)) in cylindrical coordinates where we shall assume

$$\boldsymbol{u} = u_{\mathrm{r}} \boldsymbol{e}_{\mathrm{r}}, \tag{C.12}$$

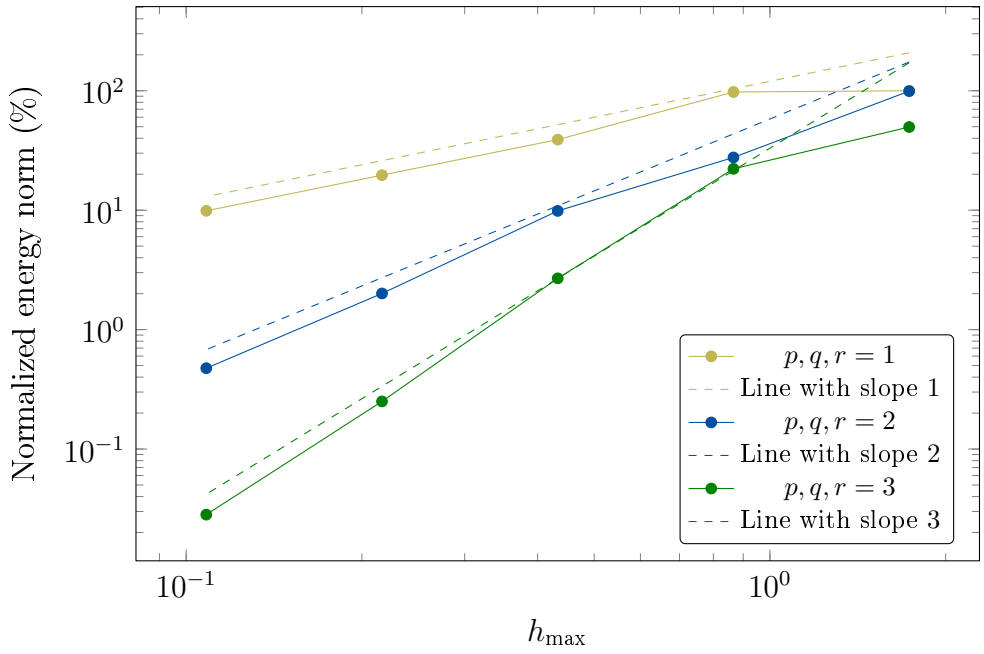


Figure C.13: Rectangular prism 4: Convergence plot.

where $u_r = u_r(r, \theta)$. That is, we only look for displacement solutions which are purely radial and independent of the z direction. Using Equation (B.17) we therefore get

$$\Delta \boldsymbol{u} = \left(\Delta u_r - \frac{1}{r^2} u_r \right) \boldsymbol{e}_r + \frac{2}{r^2} \frac{\partial u_r}{\partial \theta} \boldsymbol{e}_\theta.$$

Moreover, using Equation (B.16) we get

$$\nabla \cdot \boldsymbol{u} = \frac{1}{r} \frac{\partial (r u_r)}{\partial r},$$

such that (using Equation (B.14)) we get

$$\nabla(\nabla \cdot \boldsymbol{u}) = \frac{\partial}{\partial r} \left(\frac{1}{r} \frac{\partial (r u_r)}{\partial r} \right) \boldsymbol{e}_r + \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{1}{r} \frac{\partial (r u_r)}{\partial r} \right) \boldsymbol{e}_\theta.$$

The θ-component of Equation (C.11) is thus given by

$$\mu \frac{2}{r^2} \frac{\partial u_r}{\partial \theta} + (\lambda + \mu) \frac{1}{r} \frac{\partial}{\partial \theta} \left(\frac{1}{r} \frac{\partial (r u_r)}{\partial r} \right) = 0.$$

Assuming the solution to take the form $u_r = F(r)G(\theta)$ (separation of variables) we get

$$2\mu F(r) G'(\theta) + (\lambda + \mu) G'(\theta) \frac{\mathrm{d}(rF(r))}{\mathrm{d}r} = 0$$

If $G'(\theta) = 0$ the solution u_r is only depending on the radial variable, and this equation is trivially satisfied. We shall later consider this case. Assume now that $G'(\theta) \neq 0$. Then, we find the solution

$$F(r) = C r^{4\nu - 3}. \tag{C.13}$$

The r-component of Equation (C.11) is given by

$$\mu\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_{\mathrm{r}}}{\partial r}\right)+\frac{1}{r^2}\frac{\partial^2 u_{\mathrm{r}}}{\partial\theta^2}-\frac{1}{r^2}u_{\mathrm{r}}\right)+(\lambda+\mu)\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial(ru_{\mathrm{r}})}{\partial r}\right)=0, \tag{C.14}$$

where we have expanded the expression of Δu_{r} in terms of polar coordinates using Equation (B.14). Inserting the solution Equation (C.13) into this equation yields

$$G''(\theta)+[4(1-\nu)]^2G(\theta)=0,$$

which result in trigonometric solutions. The final solution is thus

$$u_{\mathrm{r}}(r,\theta)=r^{4\nu-3}\left(C_1\cos\left[4(1-\nu)\theta\right]+C_2\sin\left[4(1-\nu)\theta\right]\right).$$

One typically want the periodicity condition $u_{\mathrm{r}}(r,0)=u_{\mathrm{r}}(r,2\pi n)$ to hold. But such a condition will add a constraint on the parameter ν. This is not of our interest, and the solution is thus disregarded.

Let's now consider the case $G'(\theta)=0$, which corresponds to a purely radial displacement field which is only depending on the radius. That is $u_{\mathrm{r}}=u_{\mathrm{r}}(r)$. We may then without loss of generality write $u_{\mathrm{r}}=F(r)$. Equation Equation (C.14) then reduces to

$$\mu\left(\frac{1}{r}\frac{\mathrm{d}}{\mathrm{d}r}\left(rF'(r)\right)-\frac{1}{r^2}F(r)\right)+(\lambda+\mu)\frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r}\frac{\mathrm{d}(rF(r))}{\mathrm{d}r}\right)=0.$$

By expanding (and using the fact that $\lambda+2\mu\neq 0$) we get the equation

$$F''(r)+\frac{1}{r}F'(r)-\frac{1}{r^2}F(r)=0,$$

which admits solution of the form

$$F(r)=C_1r+\frac{C_2}{r}.$$

The finale solution is thus simply

$$u_{\mathrm{r}}(r)=C_1r+\frac{C_2}{r}.$$

C.5.1 Solid circular cylinder

Consider now the Solid circular cylinder subject to internal and/or external pressure. These condition are given by

$$\sigma_{\mathrm{rr}}(R_{\mathrm{o}})=-p_{\mathrm{o}}\quad\text{and}\quad\sigma_{\mathrm{rr}}(R_{\mathrm{i}})=-p_{\mathrm{i}}.$$

From Equation (B.19), we get $\varepsilon_{\mathrm{zz}}=0$,

$$\varepsilon_{\mathrm{rr}}=\frac{\partial u_{\mathrm{r}}}{\partial r}=C_1-\frac{C_2}{r^2}$$

and

$$\varepsilon_{\theta\theta}=\frac{1}{r}\left(\frac{\partial u_\theta}{\partial\theta}+u_{\mathrm{r}}\right)=C_1+\frac{C_2}{r^2},$$

such that (using Equation (B.18))

$$\sigma_{\mathrm{rr}} = (\lambda + 2\mu)\varepsilon_{\mathrm{rr}} + \lambda\varepsilon_{\theta\theta} + \lambda\varepsilon_{\mathrm{zz}} = 2(\lambda + \mu)C_1 - 2\mu\frac{C_2}{r^2}.$$

Hence, we must solve

$$\begin{aligned} 2(\lambda + \mu)C_1 - 2\mu\frac{C_2}{R_{\mathrm{o}}^2} &= -p_{\mathrm{o}} \\ 2(\lambda + \mu)C_1 - 2\mu\frac{C_2}{R_{\mathrm{i}}^2} &= -p_{\mathrm{i}}, \end{aligned}$$

which yields

$$C_1 = \frac{1}{2}\frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{(\lambda + \mu)(R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2)} \quad \text{and} \quad C_2 = \frac{1}{2}\frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 (p_{\mathrm{i}} - p_{\mathrm{o}})}{\mu(R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2)}.$$

The full determined solution is thus

$$u_{\mathrm{r}}(r) = \frac{1}{2}\frac{1}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\left(\frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{\lambda + \mu} r + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2}{\mu r}(p_{\mathrm{i}} - p_{\mathrm{o}})\right).$$

with the following stress field

$$\begin{aligned} \sigma_{\mathrm{rr}} &= \frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{o}} - R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{i}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\frac{1}{r^2} \\ \sigma_{\theta\theta} &= \frac{R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} + \frac{R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{i}} - R_{\mathrm{i}}^2 R_{\mathrm{o}}^2 p_{\mathrm{o}}}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}\frac{1}{r^2} \\ \sigma_{\mathrm{zz}} &= \frac{2\nu(R_{\mathrm{i}}^2 p_{\mathrm{i}} - R_{\mathrm{o}}^2 p_{\mathrm{o}})}{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2} \\ \sigma_{\theta\mathrm{z}} &= 0 \\ \sigma_{\mathrm{rz}} &= 0 \\ \sigma_{\mathrm{r}\theta} &= 0. \end{aligned}$$

C.6 Elasticity problem in spherical coordinates

We shall here develop analytic solutions to the elasticity problem for both the static case and the dynamic case, starting with the static case.

C.6.1 The static case

We shall assume the solution of Equation (C.11) to admit purely radial solutions which only depends on the radius. That is,

$$\boldsymbol{u} = u_{\mathrm{r}}\boldsymbol{e}_{\mathrm{r}}, \tag{C.15}$$

where $u_{\mathrm{r}} = u_{\mathrm{r}}(r)$. Using Equation (B.29) we therefore get

$$\Delta\boldsymbol{u} = \left(\Delta u_{\mathrm{r}} - \frac{2}{r^2}u_{\mathrm{r}}\right)\boldsymbol{e}_{\mathrm{r}}.$$

Moreover, using Equation (B.16) we get

$$\nabla \cdot \boldsymbol{u} = \frac{1}{r^2}\frac{\partial(r^2 u_\mathrm{r})}{\partial r},$$

such that (using Equation (B.14)) we get

$$\nabla(\nabla \cdot \boldsymbol{u}) = \frac{\partial}{\partial r}\left(\frac{1}{r^2}\frac{\partial(r^2 u_\mathrm{r})}{\partial r}\right)\boldsymbol{e}_\mathrm{r}.$$

The r-component of Equation (C.11) is given by

$$\mu\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial u_\mathrm{r}}{\partial r}\right) - \frac{2}{r^2}u_\mathrm{r}\right) + (\lambda+\mu)\frac{\partial}{\partial r}\left(\frac{1}{r^2}\frac{\partial(r^2 u_\mathrm{r})}{\partial r}\right) = 0, \tag{C.16}$$

where we have expanded the expression of Δu_r in terms of spherical coordinates using Equation (B.23).

Let now $F(r) = u_\mathrm{r}$. Then

$$\mu\left(\frac{1}{r^2}\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2F'\right) - \frac{2}{r^2}F\right) + (\lambda+\mu)\frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r^2}\frac{\mathrm{d}(r^2F)}{\mathrm{d}r}\right) = 0,$$

which reduces to

$$F'' + \frac{2}{r}F' - \frac{2}{r^2}F = 0.$$

and admits solutions of the form

$$F(r) = C_1 r + \frac{C_2}{r^2}.$$

We now want to determine the constants C_1 and C_2 using the following boundary conditions

$$\sigma_{\mathrm{rr}}(R_\mathrm{o}) = -p_\mathrm{o} \quad \text{and} \quad \sigma_{\mathrm{rr}}(R_\mathrm{i}) = -p_\mathrm{i}.$$

From Equation (B.31), we get

$$\varepsilon_{\mathrm{rr}} = \frac{\partial u_\mathrm{r}}{\partial r} = C_1 - \frac{2C_2}{r^3},$$

$$\varepsilon_{\theta\theta} = \frac{1}{r}u_\mathrm{r} = C_1 + \frac{C_2}{r^3},$$

and

$$\varepsilon_{\phi\phi} = \frac{1}{r}u_\mathrm{r} = C_1 + \frac{C_2}{r^3},$$

such that (using Equation (B.30))

$$\begin{aligned}
\sigma_{\mathrm{rr}} &= (\lambda+2\mu)\varepsilon_{\mathrm{rr}} + \lambda\varepsilon_{\theta\theta} + \lambda\varepsilon_{\phi\phi} \\
&= (\lambda+2\mu)\left(C_1 - \frac{2C_2}{r^3}\right) + 2\lambda\left(C_1 + \frac{C_2}{r^3}\right) \\
&= (3\lambda+2\mu)C_1 - \frac{4\mu}{r^3}C_2.
\end{aligned}$$

Hence, we must solve

$$(3\lambda + 2\mu)C_1 - \frac{4\mu}{R_\mathrm{o}^3}C_2 = -p_\mathrm{o}$$
$$(3\lambda + 2\mu)C_1 - \frac{4\mu}{R_\mathrm{i}^3}C_2 = -p_\mathrm{i},$$

which yields

$$C_1 = \frac{R_\mathrm{i}^3 p_\mathrm{i} - R_\mathrm{o}^3 p_\mathrm{o}}{(3\lambda + 2\mu)(R_\mathrm{o}^3 - R_\mathrm{i}^3)} \quad \text{and} \quad C_2 = \frac{1}{4}\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{\mu(R_\mathrm{o}^3 - R_\mathrm{i}^3)}.$$

The full determined solution is thus

$$u_\mathrm{r}(r) = \frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 p_\mathrm{i} - R_\mathrm{o}^3 p_\mathrm{o}}{3\lambda + 2\mu} r + \frac{1}{4}\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{\mu r^2}\right).$$

with the following stress field

$$\begin{aligned}
\sigma_{\mathrm{rr}} &= \frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{o} - p_\mathrm{i})}{r^3} + R_\mathrm{i}^3 p_\mathrm{i} - R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\theta\theta} &= \frac{1}{2}\frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{r^3} + 2R_\mathrm{i}^3 p_\mathrm{i} - 2R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\phi\phi} &= \frac{1}{2}\frac{1}{R_\mathrm{o}^3 - R_\mathrm{i}^3}\left(\frac{R_\mathrm{i}^3 R_\mathrm{o}^3 (p_\mathrm{i} - p_\mathrm{o})}{r^3} + 2R_\mathrm{i}^3 p_\mathrm{i} - 2R_\mathrm{o}^3 p_\mathrm{o}\right) \\
\sigma_{\theta\phi} &= 0 \\
\sigma_{\mathrm{r}\phi} &= 0 \\
\sigma_{\mathrm{r}\theta} &= 0.
\end{aligned}$$

C.6.2 The dynamic case

Recall from Equation (C.4) that the dynamic case of the linear elasticity problem is given by

$$\mu\Delta\boldsymbol{u} + (\lambda + \mu)\nabla(\nabla \cdot \boldsymbol{u}) + \rho_\mathrm{s}\omega^2\boldsymbol{u} = \boldsymbol{0}.$$

The r-component of this equation is now given by

$$\mu\left(\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial u_\mathrm{r}}{\partial r}\right) - \frac{2}{r^2}u_\mathrm{r}\right) + (\lambda + \mu)\frac{\partial}{\partial r}\left(\frac{1}{r^2}\frac{\partial(r^2 u_\mathrm{r})}{\partial r}\right) + \rho_\mathrm{s}\omega^2 u_\mathrm{r} = 0. \tag{C.17}$$

Let now $F(r) = u_\mathrm{r}$. Then

$$\mu\left(\frac{1}{r^2}\frac{\mathrm{d}}{\mathrm{d}r}\left(r^2 F'\right) - \frac{2}{r^2}F\right) + (\lambda + \mu)\frac{\mathrm{d}}{\mathrm{d}r}\left(\frac{1}{r^2}\frac{\mathrm{d}(r^2 F)}{\mathrm{d}r}\right) + \rho_\mathrm{s}\omega^2 F = 0,$$

which reduces to

$$F'' + \frac{2}{r}F' + \left(\rho_\mathrm{s}\omega^2 - \frac{2}{r^2}\right)F = 0.$$

Using a simple scaling (with $x = \sqrt{rho_s}\omega r$) this is actually the spherical Bessel equation (with $n = 1$). The solution is thus

$$F(r) = C_1 j_1(\sqrt{\rho_\mathrm{s}}\omega r) + C_2 y_1(\sqrt{\rho_\mathrm{s}}\omega r).$$

We now want to determine the constants C_1 and C_2 using the following boundary conditions

$$\sigma_{\mathrm{rr}}(R_{\mathrm{o}}) = -p_{\mathrm{o}} \quad \text{and} \quad \sigma_{\mathrm{rr}}(R_{\mathrm{i}}) = -p_{\mathrm{i}}.$$

From Equation (B.31), we get

$$\varepsilon_{\mathrm{rr}} = \frac{\partial u_{\mathrm{r}}}{\partial r} = C_1\sqrt{\rho_{\mathrm{s}}}\omega j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2\sqrt{\rho_{\mathrm{s}}}\omega y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r),$$

$$\varepsilon_{\theta\theta} = \frac{1}{r}u_{\mathrm{r}} = \frac{C_1}{r}j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + \frac{C_2}{r}y_1(\sqrt{\rho_{\mathrm{s}}}\omega r),$$

and

$$\varepsilon_{\phi\phi} = \frac{1}{r}u_{\mathrm{r}} = \frac{C_1}{r}j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + \frac{C_2}{r}y_1(\sqrt{\rho_{\mathrm{s}}}\omega r),$$

such that (using Equation (B.30))

$$\begin{aligned}
\sigma_{\mathrm{rr}} &= (\lambda + 2\mu)\varepsilon_{\mathrm{rr}} + \lambda\varepsilon_{\theta\theta} + \lambda\varepsilon_{\phi\phi} \\
&= (\lambda + 2\mu)\left(C_1\sqrt{\rho_{\mathrm{s}}}\omega j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2\sqrt{\rho_{\mathrm{s}}}\omega y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r)\right) \\
&\quad + 2\lambda\left(\frac{C_1}{r}j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + \frac{C_2}{r}y_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right) \\
&= \left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + \frac{2\lambda}{r}j_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right]C_1 \\
&\quad + \left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + \frac{2\lambda}{r}y_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right]C_2.
\end{aligned}$$

Hence, we must solve

$$\begin{aligned}
&\left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega j_1'(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{o}}) + \frac{2\lambda}{R_{\mathrm{o}}}j_1(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{o}})\right]C_1 \\
&\quad + \left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega y_1'(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{o}}) + \frac{2\lambda}{R_{\mathrm{o}}}y_1(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{o}})\right]C_2 = -p_{\mathrm{o}} \\
&\left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega j_1'(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{i}}) + \frac{2\lambda}{R_{\mathrm{i}}}j_1(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{i}})\right]C_1 \\
&\quad + \left[(\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega y_1'(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{i}}) + \frac{2\lambda}{R_{\mathrm{i}}}y_1(\sqrt{\rho_{\mathrm{s}}}\omega R_{\mathrm{i}})\right]C_2 = -p_{\mathrm{i}}.
\end{aligned}$$

for the constants C_1 and C_2. For brevity we write the full solution as

$$u_{\mathrm{r}}(r) = C_1 j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1(\sqrt{\rho_{\mathrm{s}}}\omega r).$$

with the following stress field

$$\sigma_{\mathrm{rr}} = (\lambda + 2\mu)\sqrt{\rho_{\mathrm{s}}}\omega\left(C_1 j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r)\right) + \frac{2\lambda\left(C_1 j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right)}{r}$$

$$\sigma_{\theta\theta} = \lambda\sqrt{\rho_{\mathrm{s}}}\omega\left(C_1 j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r)\right) + \frac{(2\lambda + 2\mu)\left(C_1 j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right)}{r}$$

$$\sigma_{\phi\phi} = \lambda\sqrt{\rho_{\mathrm{s}}}\omega\left(C_1 j_1'(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1'(\sqrt{\rho_{\mathrm{s}}}\omega r)\right) + \frac{(2\lambda + 2\mu)\left(C_1 j_1(\sqrt{\rho_{\mathrm{s}}}\omega r) + C_2 y_1(\sqrt{\rho_{\mathrm{s}}}\omega r)\right)}{r}$$

$$\sigma_{\theta\phi} = 0$$

$$\sigma_{\mathrm{r}\phi} = 0$$

$$\sigma_{\mathrm{r}\theta} = 0.$$

Appendix D

Data for NURBS geometries

In this appendix the data for NURBS volumes will be presented. For consistency we shall always let ζ run through the thickness of the geometries. We often place axisymmetric objects along the z-axis, but as some models are initially aligned with another axis (for instance model 3), we shall use this convention for consistency. In the analysis of infinite elements using prolate spheroidal coordinate system, one needs to have the object aligned with the z-axis. This is done by a simple transformation in the physical coordinates. In Figure D.3, some additional geometries are depicted. As the analysis of these cases have been omitted from the thesis, so has the data sets. The data sets may be given on request.

D.1 Solid cylinder

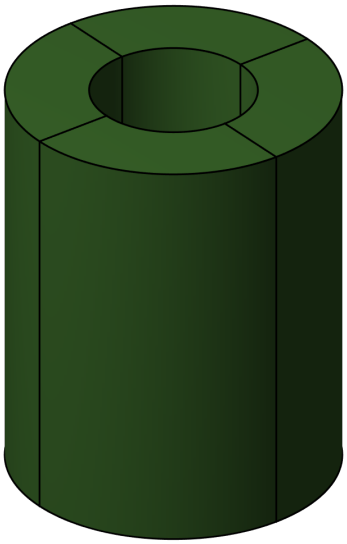
A cylinder with inner radius R_{i}, outer radius R_{o} and length L may be represented by NURBS with knot vectors tabulated in Table D.1 and control points tabulated in Table D.2. Note that we let the length of the cylinder be aligned with the z-axis. An example of this generic geometry is illustrated in Figure D.1a.

D.2 Quarter of a hemisphere

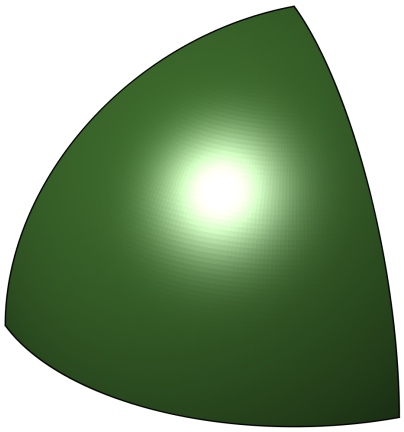
A quarter of a hemisphere with inner radius R_{i} and outer radius R_{o} may be represented by NURBS with knot vectors tabulated in Table D.3 and control points tabulated in Table D.4. The singularity points (of the mapping) is placed on the z-axis such that ξ traverses the azimuth angle and η traverses the polar angle. An example of this generic geometry is illustrated in Figure D.1b.

Table D.1: Solid cylinder: Knot vectors.

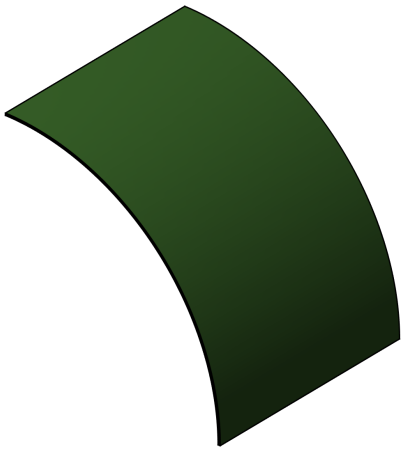
Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$
η	$q = 1$	$\mathcal{H} = \{0, 0, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$



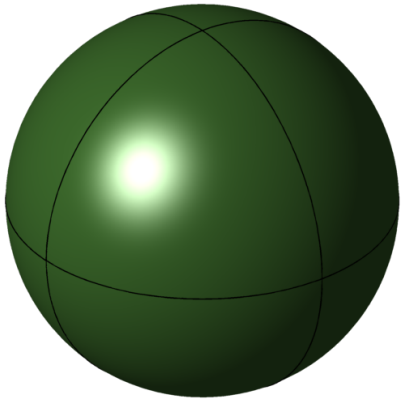
(a) **Solid cylinder:** $R_\mathrm{i} = 1$, $R_\mathrm{o} = 2$ and $L = 5$.



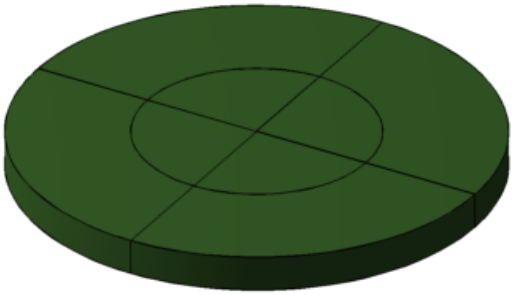
(b) **Quarter of a hemisphere:** $R_\mathrm{i} = 9.96$ and $R_\mathrm{o} = 10.04$.



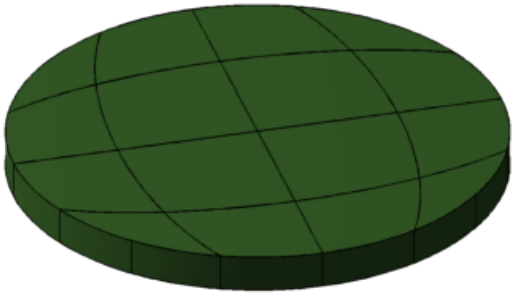
(c) **Quarter of a cylinder:** $R_\mathrm{i} = 298.5$, $R_\mathrm{o} = 301.5$ and $L = 600/2$.



(d) **Spherical shell:** $R_\mathrm{i} = 4.925$ and $R_\mathrm{o} = 5.075$.

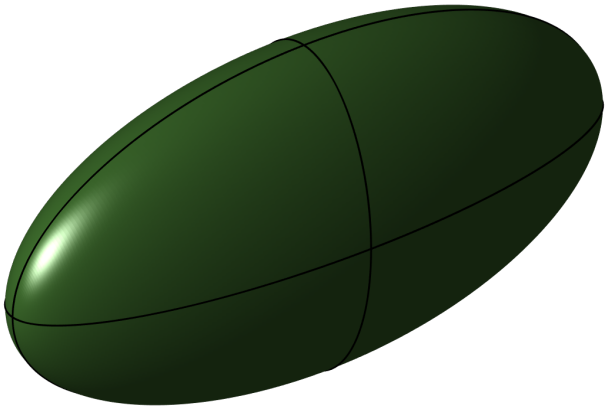


(e) **Parametrization 1 of circular plate:** $R = 2$ and $t = 0.3$.

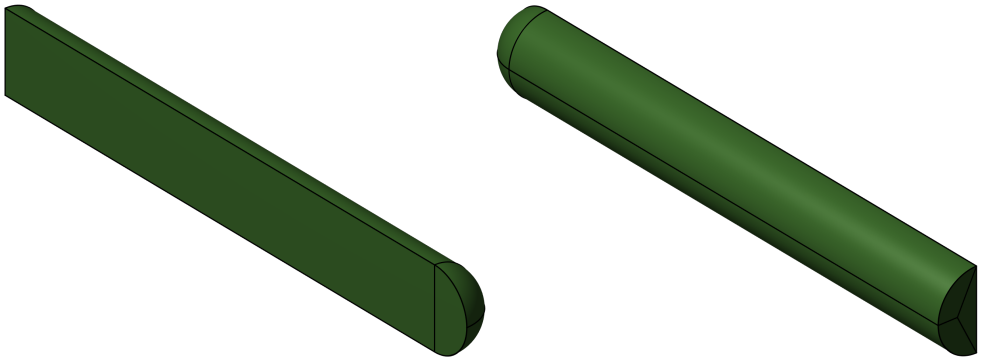


(f) **Parametrization 2 of circular plate:** $R = 2$ and $t = 0.3$.

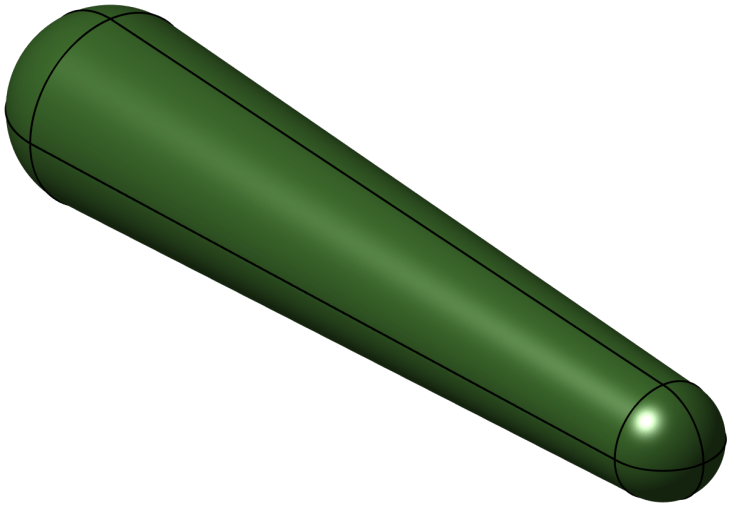
Figure D.1: NURBS geometries.



(a) **Ellipsoid:** $c_x = 4$, $c_y = 1$ and $c_z = 2$.

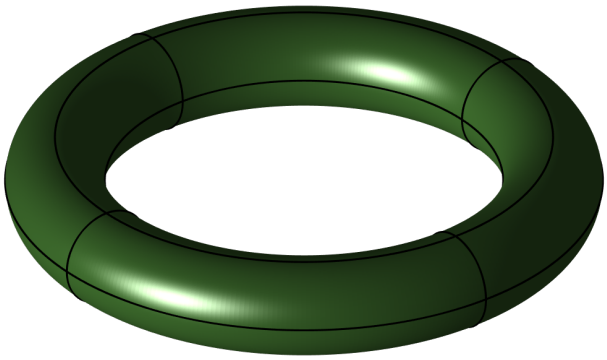


(b) **BeTSSi Model 1:** $L = 40$, $t = 0.02$ and $R_o = 3$.

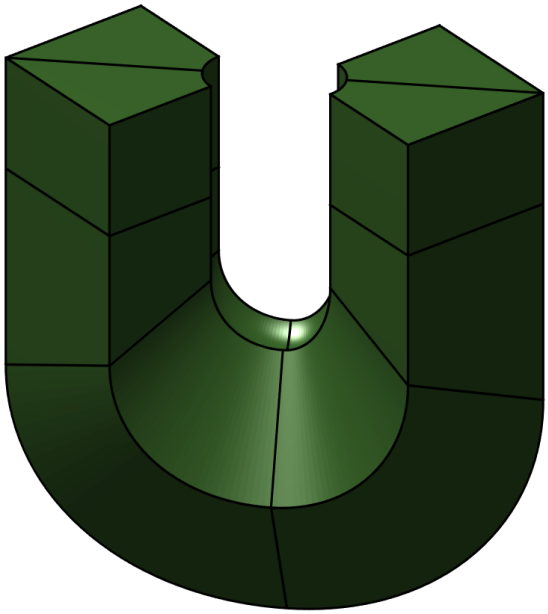


(c) **BeTSSi Model 3:** $L = 41$, $t = 0.008$, $R_{o1} = 5$ and $R_{o2} = 3$.

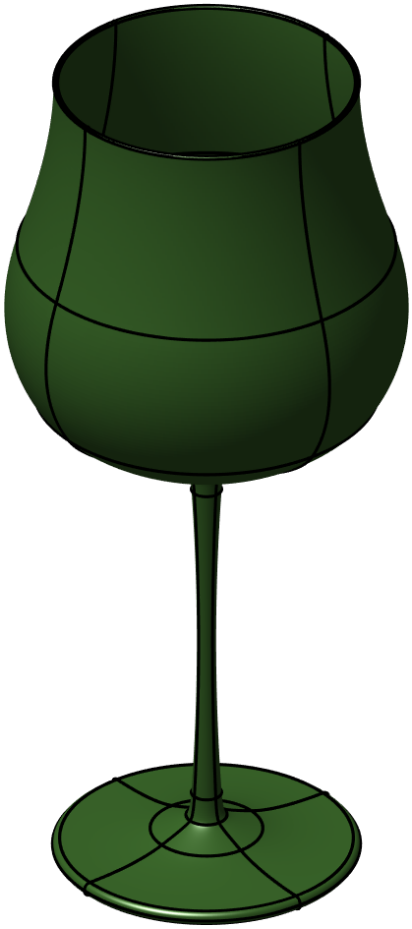
Figure D.2: NURBS geometries.



(a) A torus.



(b) The “Horse shoe”.



(c) A wine glass.

Figure D.3: NURBS geometries.

Table D.2: Solid cylinder: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
2	1	$(R_\mathrm{i}, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
4	1	$(-R_\mathrm{i}, R_\mathrm{i}, 0)$	$(-R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
6	1	$(-R_\mathrm{i}, -R_\mathrm{i}, 0)$	$(-R_\mathrm{o}, -R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	1	$(0, -R_\mathrm{i}, 0)$	$(0, -R_\mathrm{o}, 0)$	1	1
8	1	$(R_\mathrm{i}, -R_\mathrm{i}, 0)$	$(R_\mathrm{o}, -R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	1	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
1	2	$(R_\mathrm{i}, 0, L)$	$(R_\mathrm{o}, 0, L)$	1	1
2	2	$(R_\mathrm{i}, R_\mathrm{i}, L)$	$(R_\mathrm{o}, R_\mathrm{o}, L)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	2	$(0, R_\mathrm{i}, L)$	$(0, R_\mathrm{o}, L)$	1	1
4	2	$(-R_\mathrm{i}, R_\mathrm{i}, L)$	$(-R_\mathrm{o}, R_\mathrm{o}, L)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	2	$(-R_\mathrm{i}, 0, L)$	$(-R_\mathrm{o}, 0, L)$	1	1
6	2	$(-R_\mathrm{i}, -R_\mathrm{i}, L)$	$(-R_\mathrm{o}, -R_\mathrm{o}, L)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	2	$(0, -R_\mathrm{i}, L)$	$(0, -R_\mathrm{o}, L)$	1	1
8	2	$(R_\mathrm{i}, -R_\mathrm{i}, L)$	$(R_\mathrm{o}, -R_\mathrm{o}, L)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	2	$(R_\mathrm{i}, 0, L)$	$(R_\mathrm{o}, 0, L)$	1	1

Table D.3: Quarter of a hemisphere: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 1\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, 1, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

Table D.4: Quarter of a hemisphere: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
2	1	$(R_\mathrm{i}, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
1	2	$(R_\mathrm{i}, 0, R_\mathrm{i})$	$(R_\mathrm{o}, 0, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
2	2	$(R_\mathrm{i}, R_\mathrm{i}, R_\mathrm{i})$	$(R_\mathrm{o}, R_\mathrm{o}, R_\mathrm{o})$	$1/2$	$1/2$
3	2	$(0, R_\mathrm{i}, R_\mathrm{i})$	$(0, R_\mathrm{o}, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	3	$(0, 0, R_\mathrm{i})$	$(0, 0, R_\mathrm{o})$	1	1
2	3	$(0, 0, R_\mathrm{i})$	$(0, 0, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	3	$(0, 0, R_\mathrm{i})$	$(0, 0, R_\mathrm{o})$	1	1

Table D.5: Part of a cylinder: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 1\}$
η	$q = 1$	$\mathcal{H} = \{0, 0, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

Table D.6: Part of a cylinder: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(x_{\mathrm{i}1}, 0, z_{\mathrm{i}1})$	$(x_{\mathrm{o}1}, 0, z_{\mathrm{o}1})$	1	1
2	1	$(x_{\mathrm{i}2}, 0, z_{\mathrm{i}2})$	$(x_{\mathrm{o}1}, 0, z_{\mathrm{o}2})$	w	w
3	1	$(x_{\mathrm{i}3}, 0, z_{\mathrm{i}3})$	$(x_{\mathrm{o}1}, 0, z_{\mathrm{o}3})$	1	1
1	2	$(x_{\mathrm{i}1}, L, z_{\mathrm{i}1})$	$(x_{\mathrm{o}1}, L, z_{\mathrm{o}1})$	1	1
2	2	$(x_{\mathrm{i}2}, L, z_{\mathrm{i}2})$	$(x_{\mathrm{o}1}, L, z_{\mathrm{o}2})$	w	w
3	2	$(x_{\mathrm{i}3}, L, z_{\mathrm{i}3})$	$(x_{\mathrm{o}1}, L, z_{\mathrm{o}3})$	1	1

D.3 Part of a cylinder

A part of a cylinder which spans the polar angles $[\theta, \theta + \phi]$ of a cylinder with inner radius R_{i}, outer radius R_{o} and length L may be represented by NURBS with knot vectors tabulated in Table D.5 and control points tabulated in Table D.6. The parametrization starts at the polar angle θ and ends at $\theta + \phi$. The following notations has been used

$$
\begin{aligned}
x_{\mathrm{i}1} &= R_{\mathrm{i}} \cos\theta, & x_{\mathrm{i}2} &= R_{\mathrm{i}} \left(\cos\theta - \sin\theta \tan\frac{\phi}{2} \right), & x_{\mathrm{i}3} &= -R_{\mathrm{i}} \cos(\phi + \theta) \\
x_{\mathrm{o}1} &= R_{\mathrm{o}} \cos\theta, & x_{\mathrm{o}2} &= R_{\mathrm{o}} \left(\cos\theta - \sin\theta \tan\frac{\phi}{2} \right), & x_{\mathrm{o}3} &= -R_{\mathrm{o}} \cos(\phi + \theta) \\
z_{\mathrm{i}1} &= R_{\mathrm{i}} \sin\theta, & z_{\mathrm{i}2} &= R_{\mathrm{i}} \left(\sin\theta + \cos\theta \tan\frac{\phi}{2} \right), & z_{\mathrm{i}3} &= R_{\mathrm{i}} \sin(\phi + \theta) \\
z_{\mathrm{o}1} &= R_{\mathrm{o}} \sin\theta, & z_{\mathrm{o}2} &= R_{\mathrm{o}} \left(\sin\theta + \cos\theta \tan\frac{\phi}{2} \right), & z_{\mathrm{o}3} &= R_{\mathrm{o}} \sin(\phi + \theta)
\end{aligned}
$$

and the nontrivial weight is calculated by

$$
w = \cos\frac{\phi}{2}.
$$

Note that we let the length of the cylinder be aligned with the y-axis. An example of this generic geometry is illustrated in Figure D.1c where $\phi = \frac{\pi}{2}$ and $\theta = 0$.

D.4 Circular plate

A circular plate with radius R and thickness t may be represented by NURBS with knot vectors tabulated in Table D.7 and control points tabulated in Table D.10 (refered to as parametrization 1). An other parametrization of the same geometry has knot

Table D.7: Circular plate parametrization 1: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$
η	$q = 1$	$\mathcal{H} = \{0, 0, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

Table D.8: Circular plate parametrization 1: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(0, 0, 0)$	$(0, 0, t)$	1	1
2	1	$(0, 0, 0)$	$(0, 0, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(0, 0, 0)$	$(0, 0, t)$	1	1
4	1	$(0, 0, 0)$	$(0, 0, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	1	$(0, 0, 0)$	$(0, 0, t)$	1	1
6	1	$(0, 0, 0)$	$(0, 0, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	1	$(0, 0, 0)$	$(0, 0, t)$	1	1
8	1	$(0, 0, 0)$	$(0, 0, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	1	$(0, 0, 0)$	$(0, 0, t)$	1	1
1	2	$(R, 0, 0)$	$(R, 0, t)$	1	1
2	2	$(R, R, 0)$	(R, R, t)	$1/\sqrt{2}$	$1/\sqrt{2}$
3	2	$(0, R, 0)$	$(0, R, t)$	1	1
4	2	$(-R, R, 0)$	$(-R, R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	2	$(-R, 0, 0)$	$(-R, 0, t)$	1	1
6	2	$(-R, -R, 0)$	$(-R, -R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	2	$(0, -R, 0)$	$(0, -R, t)$	1	1
8	2	$(R, -R, 0)$	$(R, -R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	2	$(R, 0, 0)$	$(R, 0, t)$	1	1

vectors tabulated in Table D.9 and control points tabulated in Table D.8 (refered to as parametrization 2). From the second parametrization we see that such a plate (or the topological identical cylinder) may be reprecented by one NURBS element. Note that we let the thickness of the disk be aligned with the z-axis. Also note that (for visualization purposes) the knot $\eta = 0.5$ has been inserted in Figure D.1e and the knots $\xi = 0.25, 0.5, 0.75$ and $\eta = 0.25, 0.5, 0.75$ has been inserted in Figure D.1f. A central difference between these two parametrizations is the location of the singularity of the mapping. In parametrization 1, there is singularities along the axisymmetric axis, while the singularities of the second parametrization is located at the four combinations of $\xi = 0, 1$ and $\eta = 0, 1$.

Table D.9: Circular plate parametrization 2: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 1\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, 1, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

Table D.10: Circular plate parametrization 2: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(R, 0, 0)$	$(R, 0, t)$	1	1
2	1	$(R, -R, 0)$	$(R, -R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(0, -R, 0)$	$(0, -R, t)$	1	1
1	2	$(R, R, 0)$	(R, R, t)	$1/\sqrt{2}$	$1/\sqrt{2}$
2	2	$(0, 0, 0)$	$(0, 0, t)$	$\sqrt{2}-1$	$\sqrt{2}-1$
3	2	$(-R, -R, 0)$	$(-R, -R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	3	$(0, R, 0)$	$(0, R, t)$	1	1
2	3	$(-R, R, 0)$	$(-R, R, t)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	3	$(-R, 0, 0)$	$(-R, 0, t)$	1	1

Table D.11: Spherical shell: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, 1, 1, 2, 2, 2\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

D.5 Spherical shell

A spherical shell with inner radius R_i and outer radius R_o may be represented by NURBS with knot vectors tabulated in Table D.11 and control points tabulated in Table D.12. The singularity points (of the mapping) is placed on the x-axis such that ξ traverses around this x-axis and η traverses around the y-axis. Since ξ traverses angles from 0 to 2π, η only need to traverse angles from 0 to π (and not 2π). An example of this generic geometry is illustrated in Figure D.1d.

Table D.12: Spherical shell: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
2	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
4	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
6	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
8	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	1	$(-R_\mathrm{i}, 0, 0)$	$(-R_\mathrm{o}, 0, 0)$	1	1
1	2	$(-R_\mathrm{i}, R_\mathrm{i}, 0)$	$(-R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
2	2	$(-R_\mathrm{i}, R_\mathrm{i}, R_\mathrm{i})$	$(-R_\mathrm{o}, R_\mathrm{o}, R_\mathrm{o})$	$1/2$	$1/2$
3	2	$(-R_\mathrm{i}, 0, R_\mathrm{i})$	$(-R_\mathrm{o}, 0, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
4	2	$(-R_\mathrm{i}, -R_\mathrm{i}, R_\mathrm{i})$	$(-R_\mathrm{o}, -R_\mathrm{o}, R_\mathrm{o})$	$1/2$	$1/2$
5	2	$(-R_\mathrm{i}, -R_\mathrm{i}, 0)$	$(-R_\mathrm{o}, -R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$

Continued on next page

Table D.12 – continued from previous page

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
6	2	$(-R_\mathrm{i}, -R_\mathrm{i}, -R_\mathrm{i})$	$(-R_\mathrm{o}, -R_\mathrm{o}, -R_\mathrm{o})$	$1/2$	$1/2$
7	2	$(-R_\mathrm{i}, 0, -R_\mathrm{i})$	$(-R_\mathrm{o}, 0, -R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
8	2	$(-R_\mathrm{i}, R_\mathrm{i}, -R_\mathrm{i})$	$(-R_\mathrm{o}, R_\mathrm{o}, -R_\mathrm{o})$	$1/2$	$1/2$
9	2	$(-R_\mathrm{i}, R_\mathrm{i}, 0)$	$(-R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	3	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
2	3	$(0, R_\mathrm{i}, R_\mathrm{i})$	$(0, R_\mathrm{o}, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	3	$(0, 0, R_\mathrm{i})$	$(0, 0, R_\mathrm{o})$	1	1
4	3	$(0, -R_\mathrm{i}, R_\mathrm{i})$	$(0, -R_\mathrm{o}, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	3	$(0, -R_\mathrm{i}, 0)$	$(0, -R_\mathrm{o}, 0)$	1	1
6	3	$(0, -R_\mathrm{i}, -R_\mathrm{i})$	$(0, -R_\mathrm{o}, -R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	3	$(0, 0, -R_\mathrm{i})$	$(0, 0, -R_\mathrm{o})$	1	1
8	3	$(0, R_\mathrm{i}, -R_\mathrm{i})$	$(0, R_\mathrm{o}, -R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	3	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
1	4	$(R_\mathrm{i}, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
2	4	$(R_\mathrm{i}, R_\mathrm{i}, R_\mathrm{i})$	$(R_\mathrm{o}, R_\mathrm{o}, R_\mathrm{o})$	$1/2$	$1/2$
3	4	$(R_\mathrm{i}, 0, R_\mathrm{i})$	$(R_\mathrm{o}, 0, R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
4	4	$(R_\mathrm{i}, -R_\mathrm{i}, R_\mathrm{i})$	$(R_\mathrm{o}, -R_\mathrm{o}, R_\mathrm{o})$	$1/2$	$1/2$
5	4	$(R_\mathrm{i}, -R_\mathrm{i}, 0)$	$(R_\mathrm{o}, -R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
6	4	$(R_\mathrm{i}, -R_\mathrm{i}, -R_\mathrm{i})$	$(R_\mathrm{o}, -R_\mathrm{o}, -R_\mathrm{o})$	$1/2$	$1/2$
7	4	$(R_\mathrm{i}, 0, -R_\mathrm{i})$	$(R_\mathrm{o}, 0, -R_\mathrm{o})$	$1/\sqrt{2}$	$1/\sqrt{2}$
8	4	$(R_\mathrm{i}, R_\mathrm{i}, -R_\mathrm{i})$	$(R_\mathrm{o}, R_\mathrm{o}, -R_\mathrm{o})$	$1/2$	$1/2$
9	4	$(R_\mathrm{i}, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
2	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
4	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
6	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
8	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	5	$(R_\mathrm{i}, 0, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1

D.6 Ellipsoid

An ellipsoid (surface) with semi principle axes c_x, c_y and c_z may be represented by NURBS with knot vectors tabulated in Table D.13 and control points tabulated in Table D.14.

The singularity points (of the mapping) is placed on the z-axis such that ξ traverses around this z-axis and η traverses around the y-axis. Since ξ traverses angles from 0 to 2π, η only need to traverse angles from 0 to π (and not 2π). An example of this generic geometry is illustrated in Figure D.1d.

Table D.13: Ellipsoid: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, 1, 1, 2, 2, 2\}$

It should be noted that when prolate spheroids are considered, we shall use $c_\mathrm{x} = c_\mathrm{y}$ such that the spheroid is aligned with the z-axis. Moreover, the parameter ξ will resemble the azimuth angle ϕ while the parameter η will resemble the polar angle θ (but will travel in the opposite direction).

It is also possible to create an ellipsoid with only three elements in η-direction (but no further reduction in the number of elements in this direction). It would then be possible to construct a more suited parametrization for the prolate spheroid when analyzing model 1. As this model was not analyzed in the thesis this parametrization has been omitted.

Table D.14: Ellipsoid: Control points and weights.

i	j	$\boldsymbol{P}_{i,j}$	$w_{i,j}$
1	1	$(0, 0, -c_\mathrm{z})$	1
2	1	$(0, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
3	1	$(0, 0, -c_\mathrm{z})$	1
4	1	$(0, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
5	1	$(0, 0, -c_\mathrm{z})$	1
6	1	$(0, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
7	1	$(0, 0, -c_\mathrm{z})$	1
8	1	$(0, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
9	1	$(0, 0, -c_\mathrm{z})$	1
1	2	$(c_\mathrm{x}, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
2	2	$(c_\mathrm{x}, c_\mathrm{y}, -c_\mathrm{z})$	$1/2$
3	2	$(0, c_\mathrm{y}, -c_\mathrm{z})$	$1/\sqrt{2}$
4	2	$(-c_\mathrm{x}, c_\mathrm{y}, -c_\mathrm{z})$	$1/2$
5	2	$(-c_\mathrm{x}, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
6	2	$(-c_\mathrm{x}, -c_\mathrm{y}, -c_\mathrm{z})$	$1/2$
7	2	$(0, -c_\mathrm{y}, -c_\mathrm{z})$	$1/\sqrt{2}$
8	2	$(c_\mathrm{x}, -c_\mathrm{y}, -c_\mathrm{z})$	$1/2$
9	2	$(c_\mathrm{x}, 0, -c_\mathrm{z})$	$1/\sqrt{2}$
1	3	$(c_\mathrm{x}, 0, 0)$	1
2	3	$(c_\mathrm{x}, c_\mathrm{y}, 0)$	$1/\sqrt{2}$
3	3	$(0, c_\mathrm{y}, 0)$	1
4	3	$(-c_\mathrm{x}, c_\mathrm{y}, 0)$	$1/\sqrt{2}$
5	3	$(-c_\mathrm{x}, 0, 0)$	1
6	3	$(-c_\mathrm{x}, -c_\mathrm{y}, 0)$	$1/\sqrt{2}$
7	3	$(0, -c_\mathrm{y}, 0)$	1
8	3	$(c_\mathrm{x}, -c_\mathrm{y}, 0)$	$1/\sqrt{2}$
9	3	$(c_\mathrm{x}, 0, 0)$	1
1	4	$(c_\mathrm{x}, 0, c_\mathrm{z})$	$1/\sqrt{2}$

Continued on next page

Table D.14 – continued from previous page

i	j	$\boldsymbol{P}_{i,j}$	$w_{i,j}$
2	4	$(c_\mathrm{x}, c_\mathrm{y}, c_\mathrm{z})$	$1/2$
3	4	$(0, c_\mathrm{y}, c_\mathrm{z})$	$1/\sqrt{2}$
4	4	$(-c_\mathrm{x}, c_\mathrm{y}, c_\mathrm{z})$	$1/2$
5	4	$(-c_\mathrm{x}, 0, c_\mathrm{z})$	$1/\sqrt{2}$
6	4	$(-c_\mathrm{x}, -c_\mathrm{y}, c_\mathrm{z})$	$1/2$
7	4	$(0, -c_\mathrm{y}, c_\mathrm{z})$	$1/\sqrt{2}$
8	4	$(c_\mathrm{x}, -c_\mathrm{y}, c_\mathrm{z})$	$1/2$
9	4	$(c_\mathrm{x}, 0, c_\mathrm{z})$	$1/\sqrt{2}$
1	5	$(0, 0, c_\mathrm{z})$	1
2	5	$(0, 0, c_\mathrm{z})$	$1/\sqrt{2}$
3	5	$(0, 0, c_\mathrm{z})$	1
4	5	$(0, 0, c_\mathrm{z})$	$1/\sqrt{2}$
5	5	$(0, 0, c_\mathrm{z})$	1
6	5	$(0, 0, c_\mathrm{z})$	$1/\sqrt{2}$
7	5	$(0, 0, c_\mathrm{z})$	1
8	5	$(0, 0, c_\mathrm{z})$	$1/\sqrt{2}$
9	5	$(0, 0, c_\mathrm{z})$	1

D.7 BeTSSi Model 1

BeTSSi Model 1 is defined with the following parameters: The outer radius of the hemispherical endcap R_o, the thickness t and the distance between the endcap and the flat end L[1]. This model may be represented by NURBS with knot vectors tabulated in Table D.15 and control points tabulated in Table D.16.

An example of this generic geometry is illustrated in Figure D.2b with parameters used in the BeTSSi community.

The following notations has been used.

- Dependent parameters for the outer surface

$$
\begin{aligned}
y_1 &= \frac{R_\mathrm{o}}{20}(8 - \sqrt{29}) \\
z_1 &= \frac{R_\mathrm{o}}{10}(2 + \sqrt{29}) \\
z_2 &= z_1 - \frac{y_1}{z_1}\left(\frac{3R_\mathrm{o}}{4} - y_1\right) \\
w_1 &= \cos\frac{\theta_1}{2}, \quad \text{where} \quad \theta_1 = \cos^{-1}\left(\frac{4y_1}{3R_\mathrm{o}}\right)
\end{aligned}
$$

[1] In the BeTSSi description, L defines the total length of the model. We here choose this other convention just for convenience.

Table D.15: BeTSSi Model 1: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 3\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, \eta_1, \eta_1, \eta_2, \eta_2, 1, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

- Dependent parameters for the inner surface

$$
\begin{aligned}
x_1 &= \sqrt{R_\mathrm{i}^2 - t^2} \\
x_2 &= x_1 - \frac{t}{x_1}(R_\mathrm{i} - t) \\
z_5 &= \sqrt{R_\mathrm{i}^2 - t^2} \\
z_6 &= z_5 - \frac{t}{z_5}(R_\mathrm{i} - t) \\
y_4 &= \frac{1}{2}\left(\frac{R_\mathrm{o}}{2} + R_\mathrm{i}\right) \\
y_3 &= \frac{-b - \sqrt{b^2 - 4ac}}{2a}, \quad \text{where} \quad a = 1 + \left(\frac{2z_5}{R_\mathrm{o} - 2t}\right)^2, \quad b = -R_\mathrm{o}\left(\frac{2z_5}{R_\mathrm{o} - 2t}\right)^2, \\
&\qquad\qquad \text{and} \quad c = \frac{R_\mathrm{o}^2}{4}\left(\frac{2z_5}{R_\mathrm{o} - 2t}\right)^2 - y_4^2 \\
z_3 &= \sqrt{y_4^2 - y_3^2} \\
z_4 &= z_3 - \frac{y_3}{z_3}(y_4 - y_3) \\
w_2 &= \cos\frac{\theta_2}{2}, \quad \text{where} \quad \theta_2 = \cos^{-1}\left(\frac{y_3}{y_4}\right) \\
w_3 &= \cos\frac{\theta_3}{2}, \quad \text{where} \quad \theta_3 = \cos^{-1}\left(\frac{t}{R_\mathrm{i}}\right).
\end{aligned}
$$

Table D.16: BeTSSi Model 1: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	1	1
2	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	w_2	w_1
3	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	1	1
4	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	1	1
5	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	1	1
6	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	w_2	w_1
7	1	$(t - L, R_\mathrm{o}/2, 0)$	$(-L, R_\mathrm{o}/2, 0)$	1	1
1	2	$(t - L, y_4, 0)$	$(-L, \frac{3}{4}R_\mathrm{o}, 0)$	1	1
2	2	$(t - L, y_4, z_4)$	$(-L, \frac{3}{4}R_\mathrm{o}, z_2)$	w_2	w_1
3	2	$(t - L, y_3, z_3)$	$(-L, y_1, z_1)$	1	1
4	2	$(t - L, y_3, 0)$	$(-L, y_1, 0)$	1	1

Continued on next page

Table D.16 – continued from previous page

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
5	2	$(t-L, y_3, -z_3)$	$(-L, y_1, -z_1)$	1	1
6	2	$(t-L, y_4, -z_4)$	$(-L, \frac{3}{4}R_\mathrm{o}, -z_2)$	w_2	w_1
7	2	$(t-L, y_4, 0)$	$(-L, \frac{3}{4}R_\mathrm{o}, 0)$	1	1
1	3	$(t-L, R_\mathrm{i}, 0)$	$(-L, R_\mathrm{o}, 0)$	1	1
2	3	$(t-L, R_\mathrm{i}, z_6)$	$(-L, R_\mathrm{o}, R_\mathrm{o})$	w_3	$1/\sqrt{2}$
3	3	$(t-L, t, z_5)$	$(-L, 0, R_\mathrm{o})$	1	1
4	3	$(t-L, t, 0)$	$(-L, 0, 0)$	1	1
5	3	$(t-L, t, -z_5)$	$(-L, 0, -R_\mathrm{o})$	1	1
6	3	$(t-L, R_\mathrm{i}, -z_6)$	$(-L, R_\mathrm{o}, -R_\mathrm{o})$	w_3	$1/\sqrt{2}$
7	3	$(t-L, R_\mathrm{i}, 0)$	$(-L, R_\mathrm{o}, 0)$	1	1
1	4	$(-L/2, R_\mathrm{i}, 0)$	$(-L/2, R_\mathrm{o}, 0)$	1	1
2	4	$(-L/2, R_\mathrm{i}, z_6)$	$(-L/2, R_\mathrm{o}, R_\mathrm{o})$	w_3	$1/\sqrt{2}$
3	4	$(-L/2, t, z_5)$	$(-L/2, 0, R_\mathrm{o})$	1	1
4	4	$(-L/2, t, 0)$	$(-L/2, 0, 0)$	1	1
5	4	$(-L/2, t, -z_5)$	$(-L/2, 0, -R_\mathrm{o})$	1	1
6	4	$(-L/2, R_\mathrm{i}, -z_6)$	$(-L/2, R_\mathrm{o}, -R_\mathrm{o})$	w_3	$1/\sqrt{2}$
7	4	$(-L/2, R_\mathrm{i}, 0)$	$(-L/2, R_\mathrm{o}, 0)$	1	1
1	5	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
2	5	$(0, R_\mathrm{i}, z_6)$	$(0, R_\mathrm{o}, R_\mathrm{o})$	w_3	$1/\sqrt{2}$
3	5	$(0, t, z_5)$	$(0, 0, R_\mathrm{o})$	1	1
4	5	$(0, t, 0)$	$(0, 0, 0)$	1	1
5	5	$(0, t, -z_5)$	$(0, 0, -R_\mathrm{o})$	1	1
6	5	$(0, R_\mathrm{i}, -z_6)$	$(0, R_\mathrm{o}, -R_\mathrm{o})$	w_3	$1/\sqrt{2}$
7	5	$(0, R_\mathrm{i}, 0)$	$(0, R_\mathrm{o}, 0)$	1	1
1	6	$(x_2, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	w_3	$1/\sqrt{2}$
2	6	(x_2, R_i, z_6)	$(R_\mathrm{o}, R_\mathrm{o}, R_\mathrm{o})$	w_3^2	1/2
3	6	(x_2, t, z_5)	$(R_\mathrm{o}, 0, R_\mathrm{o})$	w_3	$1/\sqrt{2}$
4	6	$(x_2, t, 0)$	$(R_\mathrm{o}, 0, 0)$	w_3	1
5	6	$(x_2, t, -z_5)$	$(R_\mathrm{o}, 0, -R_\mathrm{o})$	w_3	$1/\sqrt{2}$
6	6	$(x_2, R_\mathrm{i}, -z_6)$	$(R_\mathrm{o}, R_\mathrm{o}, -R_\mathrm{o})$	w_3^2	1/2
7	6	$(x_2, R_\mathrm{i}, 0)$	$(R_\mathrm{o}, R_\mathrm{o}, 0)$	w_3	$1/\sqrt{2}$
1	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
2	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	w_3	$1/\sqrt{2}$
3	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
4	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
5	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1
6	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	w_3	$1/\sqrt{2}$
7	7	$(x_1, t, 0)$	$(R_\mathrm{o}, 0, 0)$	1	1

Table D.17: BeTSSi Model 3: Knot vectors.

Direction	Order	Knot vector
ξ	$p = 2$	$\Xi = \{0, 0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 4\}$
η	$q = 2$	$\mathcal{H} = \{0, 0, 0, \eta_1, \eta_1, \eta_2, \eta_2, 1, 1, 1\}$
ζ	$r = 1$	$\mathcal{Z} = \{0, 0, 1, 1\}$

D.8 BeTSSi Model 3

BeTSSi Model 3 is defined with the following parameters: The outer radii of the hemispherical endcaps $R_{\mathrm{o}1}$ and $R_{\mathrm{o}2}$, the thickness t and the distance between the endcaps L[2]. This model may be represented by NURBS with knot vectors tabulated in Table D.17 and control points tabulated in Table D.18. The following notations has been used

$$R_{\mathrm{i}1} = R_{\mathrm{o}1} - t, \quad R_{\mathrm{i}2} = R_{\mathrm{o}2} - t, \quad R_{\mathrm{im}} = \frac{R_{\mathrm{i}1} + R_{\mathrm{i}2}}{2}, \quad R_{\mathrm{om}} = \frac{R_{\mathrm{o}1} + R_{\mathrm{o}2}}{2}.$$

Note that the mock shell used in [3] is almost a special case of model 3 with $R_{\mathrm{o}1} = R_{\mathrm{o}2}$. The only difference would be that the mock shell has different thickness in the hemispheres and the cylinder while this model has uniform thickness. An example of this generic geometry is illustrated in Figure D.2c with parameters used in the BeTSSi community.

Table D.18: BeTSSi Model 3: Control points and weights.

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
1	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	1	1
2	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	1	1
4	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	1	1
6	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	1	1
8	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	1	$(-L - R_{\mathrm{i}1}, 0, 0)$	$(-L - R_{\mathrm{o}1}, 0, 0)$	1	1
1	2	$(-L - R_{\mathrm{i}1}, R_{\mathrm{i}1}, 0)$	$(-L - R_{\mathrm{o}1}, R_{\mathrm{o}1}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
2	2	$(-L - R_{\mathrm{i}1}, R_{\mathrm{i}1}, R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, R_{\mathrm{o}1}, R_{\mathrm{o}1})$	1/2	1/2
3	2	$(-L - R_{\mathrm{i}1}, 0, R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, 0, R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$
4	2	$(-L - R_{\mathrm{i}1}, -R_{\mathrm{i}1}, R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, -R_{\mathrm{o}1}, R_{\mathrm{o}1})$	1/2	1/2
5	2	$(-L - R_{\mathrm{i}1}, -R_{\mathrm{i}1}, 0)$	$(-L - R_{\mathrm{o}1}, -R_{\mathrm{o}1}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
6	2	$(-L - R_{\mathrm{i}1}, -R_{\mathrm{i}1}, -R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, -R_{\mathrm{o}1}, -R_{\mathrm{o}1})$	1/2	1/2
7	2	$(-L - R_{\mathrm{i}1}, 0, -R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, 0, -R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$
8	2	$(-L - R_{\mathrm{i}1}, R_{\mathrm{i}1}, -R_{\mathrm{i}1})$	$(-L - R_{\mathrm{o}1}, R_{\mathrm{o}1}, -R_{\mathrm{o}1})$	1/2	1/2
9	2	$(-L - R_{\mathrm{i}1}, R_{\mathrm{i}1}, 0)$	$(-L - R_{\mathrm{o}1}, R_{\mathrm{o}1}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	3	$(-L, R_{\mathrm{i}1}, 0)$	$(-L, R_{\mathrm{o}1}, 0)$	1	1
2	3	$(-L, R_{\mathrm{i}1}, R_{\mathrm{i}1})$	$(-L, R_{\mathrm{o}1}, R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$

Continued on next page

[2] In the BeTSSi description, L defines the total length of the model. We here choose this other convention just for convenience.

Table D.18 – continued from previous page

i	j	$\boldsymbol{P}_{i,j,1}$	$\boldsymbol{P}_{i,j,2}$	$w_{i,j,1}$	$w_{i,j,2}$
3	3	$(-L, 0, R_{\mathrm{i}1})$	$(-L, 0, R_{\mathrm{o}1})$	1	1
4	3	$(-L, -R_{\mathrm{i}1}, R_{\mathrm{i}1})$	$(-L, -R_{\mathrm{o}1}, R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	3	$(-L, -R_{\mathrm{i}1}, 0)$	$(-L, -R_{\mathrm{o}1}, 0)$	1	1
6	3	$(-L, -R_{\mathrm{i}1}, -R_{\mathrm{i}1})$	$(-L, -R_{\mathrm{o}1}, -R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	3	$(-L, 0, -R_{\mathrm{i}1})$	$(-L, 0, -R_{\mathrm{o}1})$	1	1
8	3	$(-L, R_{\mathrm{i}1}, -R_{\mathrm{i}1})$	$(-L, R_{\mathrm{o}1}, -R_{\mathrm{o}1})$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	3	$(-L, R_{\mathrm{i}1}, 0)$	$(-L, R_{\mathrm{o}1}, 0)$	1	1
1	4	$(-L/2, R_{\mathrm{im}}, 0)$	$(-L, R_{\mathrm{om}}, 0)$	1	1
2	4	$(-L/2, R_{\mathrm{im}}, R_{\mathrm{im}})$	$(-L, R_{\mathrm{om}}, R_{\mathrm{om}})$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	4	$(-L/2, 0, R_{\mathrm{im}})$	$(-L, 0, R_{\mathrm{om}})$	1	1
4	4	$(-L/2, -R_{\mathrm{im}}, R_{\mathrm{im}})$	$(-L, -R_{\mathrm{om}}, R_{\mathrm{om}})$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	4	$(-L/2, -R_{\mathrm{im}}, 0)$	$(-L, -R_{\mathrm{om}}, 0)$	1	1
6	4	$(-L/2, -R_{\mathrm{im}}, -R_{\mathrm{im}})$	$(-L, -R_{\mathrm{om}}, -R_{\mathrm{om}})$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	4	$(-L/2, 0, -R_{\mathrm{im}})$	$(-L, 0, -R_{\mathrm{om}})$	1	1
8	4	$(-L/2, R_{\mathrm{im}}, -R_{\mathrm{im}})$	$(-L, R_{\mathrm{om}}, -R_{\mathrm{om}})$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	4	$(-L/2, R_{\mathrm{im}}, 0)$	$(-L, R_{\mathrm{om}}, 0)$	1	1
1	5	$(0, R_{\mathrm{i}2}, 0)$	$(-L, R_{\mathrm{o}2}, 0)$	1	1
2	5	$(0, R_{\mathrm{i}2}, R_{\mathrm{i}2})$	$(-L, R_{\mathrm{o}2}, R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	5	$(0, 0, R_{\mathrm{i}2})$	$(-L, 0, R_{\mathrm{o}2})$	1	1
4	5	$(0, -R_{\mathrm{i}2}, R_{\mathrm{i}2})$	$(-L, -R_{\mathrm{o}2}, R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	5	$(0, -R_{\mathrm{i}2}, 0)$	$(-L, -R_{\mathrm{o}2}, 0)$	1	1
6	5	$(0, -R_{\mathrm{i}2}, -R_{\mathrm{i}2})$	$(-L, -R_{\mathrm{o}2}, -R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	5	$(0, 0, -R_{\mathrm{i}2})$	$(-L, 0, -R_{\mathrm{o}2})$	1	1
8	5	$(0, R_{\mathrm{i}2}, -R_{\mathrm{i}2})$	$(-L, R_{\mathrm{o}2}, -R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	5	$(0, R_{\mathrm{i}2}, 0)$	$(-L, R_{\mathrm{o}2}, 0)$	1	1
1	6	$(R_{\mathrm{i}2}, R_{\mathrm{i}2}, 0)$	$(R_{\mathrm{o}2}, R_{\mathrm{o}2}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
2	6	$(R_{\mathrm{i}2}, R_{\mathrm{i}2}, R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, R_{\mathrm{o}2}, R_{\mathrm{o}2})$	$1/2$	$1/2$
3	6	$(R_{\mathrm{i}2}, 0, R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, 0, R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
4	6	$(R_{\mathrm{i}2}, -R_{\mathrm{i}2}, R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, -R_{\mathrm{o}2}, R_{\mathrm{o}2})$	$1/2$	$1/2$
5	6	$(R_{\mathrm{i}2}, -R_{\mathrm{i}2}, 0)$	$(R_{\mathrm{o}2}, -R_{\mathrm{o}2}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
6	6	$(R_{\mathrm{i}2}, -R_{\mathrm{i}2}, -R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, -R_{\mathrm{o}2}, -R_{\mathrm{o}2})$	$1/2$	$1/2$
7	6	$(R_{\mathrm{i}2}, 0, -R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, 0, -R_{\mathrm{o}2})$	$1/\sqrt{2}$	$1/\sqrt{2}$
8	6	$(R_{\mathrm{i}2}, R_{\mathrm{i}2}, -R_{\mathrm{i}2})$	$(R_{\mathrm{o}2}, R_{\mathrm{o}2}, -R_{\mathrm{o}2})$	$1/2$	$1/2$
9	6	$(R_{\mathrm{i}2}, R_{\mathrm{i}2}, 0)$	$(R_{\mathrm{o}2}, R_{\mathrm{o}2}, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
1	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	1	1
2	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
3	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	1	1
4	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
5	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	1	1
6	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
7	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	1	1
8	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	$1/\sqrt{2}$	$1/\sqrt{2}$
9	7	$(R_{\mathrm{i}2}, 0, 0)$	$(R_{\mathrm{o}2}, 0, 0)$	1	1

Appendix E

Source code

The complete source code is far to comprehensive to be included in this thesis. However, we shall present the implementation of the NURBS routine in some detail as this is a fundamental function of the analysis. In addition we add the routine which create global matrices. We shall here not go into details. The source code, alongside instruction of how to use, may be given on request.

E.1 B-spline implementation

When we want to evaluate a B-spline at a fixed ξ it is important to note that it is very redundant to evaluate all n basis function. This is due to the small support of each function. In fact, it turns out that only (at most) $p+1$ basis functions are non-zero at ξ. It is then important to only use these function to have an efficient code. One typically implements a function which finds the knot span corresponding to a given ξ. Due to the ordering of the knot vector, we may use a binary search algorithm for finding this span. The span will be defined by the index i corresponding to the last basis function which is non-zero at ξ. The following listing represents the algorithm called `findKnotSpan` and is listed in Listing E.1.

As an example, if $p = 2$ and $n = 8$ and we want to evaluate a B-spline with the knot vector $\Xi = \{0, 0, 0, 0.1, 0.5, 0.5, 0.8, 0.9, 1, 1, 1\}$, we get the index $i = 3$ if $\xi = 0.09$, $i = 6$ if $\xi = 0.5$ and $i = n = 8$ if $\xi = 0.9$ or $\xi = 1$.

We are now ready to implement a program which evaluates B-splines using the previous routine. When the recursion formula is used to evaluate the $p + 1$ functions which are non-zero at ξ, the function $N_{i,0}$ is the only function of order zero which is non-zero at ξ. Everything is thus build from this function, such that we get the graph in Figure E.1.

It is then clear that we need two loops. The outer loop should iterate over the columns of this graph and the inner loop should iterate over the rows. The output of the function `Bspline_basis` is simply an array `N` which contains the $p + 1$ functions which are evaluated at ξ. To save memory one should also store the intermediate values $N_{i-j+k,j-1}$ in this same array. Thus, we need to store `N(k-1)` to compute `N(k)` (which we store in the variable `saved`). Here, j is the loop index of the outer loop, while k is the loop index of the inner loop. The first iteration (when $p = 0$) should be done separate.

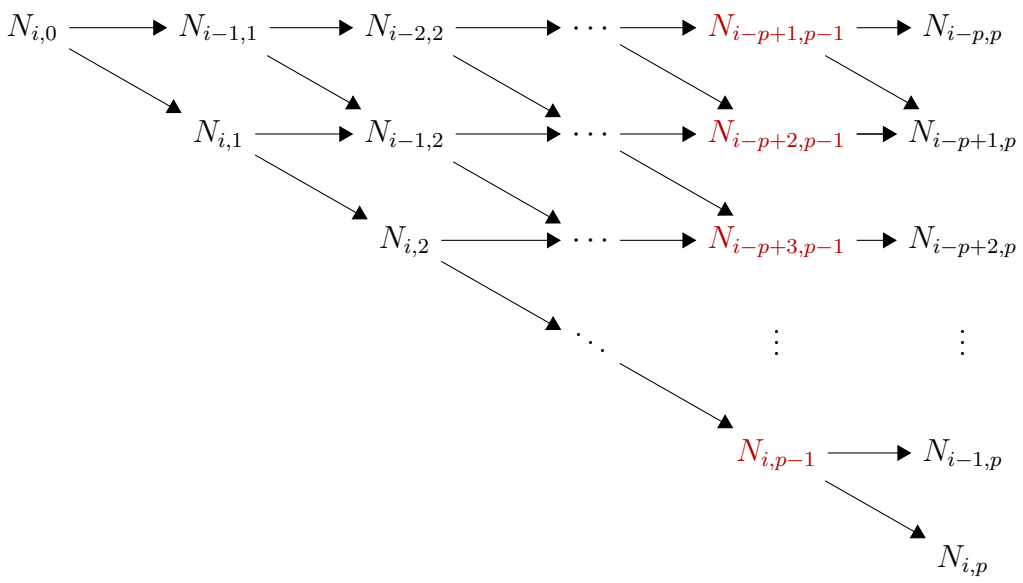


Figure E.1: B-splines evaluation graph when the $p+1$ non-zero basis functions are to be evaluated at a given ξ. The values colored red is used to calculate the nonzero derivatives at the same ξ.

We will finally need to compute the corresponding derivatives. Once again, only $p+1$ derivatives will be nonzero at ξ. Let `N_tilde` be an array containing the red values in the previous graph. The full function is found in Listing E.2.

E.2 NURBS implementation

We may now insert all of this into a routine which computes the nonzero NURBS basis functions at a given point (ξ, η, ζ) and the corresponding nonzero derivatives. This function is called `NURBS3DBasisDers`. After finding the nonzero B-spline functions and their corresponding derivatives as discussed above, it continues by first finding the function W and its corresponding derivatives.

Note the use of the index `A` which is convenient when weights is located in an long array (which is the format we shall use when doing the IGA analysis). Finally, the function can now compute the NURBS basis alongside the corresponding derivatives.

Note the use of the intermediate variables `fac` and `NML` which is used to avoid redundant computations. The index `counter` loops over the $n_{\text{en}} = (p+1)\cdot(q+1)\cdot(r+1)$ nonzero functions. The full function is found in Listing E.3.

E.3 Building global matrices

The function `buildGlobalMatrices` builds any fluid or solid matrix used in this thesis. When dynamics is considered, the mass matrix will also be computed. Finally, if any bodyforces are applied, a corresponding loading vector will be constructed. As

an example of use, consider any analysis concerning finding eigenvalues for elasticity problems. Then we use the function as follows.

```
options = {'operator', 'linearElasticity',...
           'fieldDimension', 3,...
           'buildMassMatrix', 1};
[K, M, ~] = buildGlobalMatrices(varCol, options);
```

The partial differential equation resulting in the global matrix is the linear elasticity equation. Hence, we set `'operator'` to be `'linearElasticity'`. If the Helmholtz equation was to be solved, the `'operator'` is set to be `'Laplace'`. The option `'fieldDimension'` is only added for generality if other operators where to be implemented. Indeed, the solution for the linear elasticity operator is always vector valued (in 2D/3D), while the solution for the Laplace operator is scalar valued. Finally, we ask the function to build the mass matrix, as eigenvalues concerns with dynamics. The struct `varCol` simply collects all variables associated with the NURBS mesh. The full function is listed in Listing E.4

Listing E.1: `findKnotSpan.m`

```
function i = findKnotSpan(n, p, xi, Xi)
% This routine finds the knot span corresponding to a given value xi.
% The method uses a sequencial search algorithm

% Input
%       n:    the number of control points
%       p:    the degree of the B-Spline
%       xi:   the value for which we want to find the knot span
%       Xi:   an open knot vector of size n+p+1

% Output
%       i: index of knot such that xi is an element in [Xi(i) Xi(i+1))

% Check for xi = Xi(end)
if xi == Xi(end)
    i = n;
    return;
end

low = p;
high = n + 1;
mid = floor((low + high) / 2);
while xi < Xi(mid) || xi >= Xi(mid+1)
    if xi < Xi(mid)
        high = mid;
    else
        low = mid;
    end
    mid = floor((low + high) / 2);
end

i = mid;
```

Listing E.2: Bspline_basisDers.m

```
function [N, dNdxi] = Bspline_basisDers(i, xi, p, Xi)
% This routine compute the p+1 nonzero basis functions and corresponding
% derivatives at xi

% Input
%       i:      knot span index corresonding to xi
%       p:      the degree of the B-Spline/NURBS
%       xi:     the value for which we want to evaluate the Bspline
%       Xi:     an open knot vector of size n+p+1

% Output
%       N:      array of the p+1 B-spline functions evaluated at xi




N = zeros(1,p+1);
N(1) = 1;
saved = 1;

for j = 2:p+1
    % For k = 1 there is no dependence on N(k-1) of the previous run.
    for k = 1:j
        % Compute N_{i-j+k,j-1} according to the Cox-deBoor formula
        temp = 0;
        if k ~= j
            temp = (Xi(i+k)-xi)/(Xi(i+k)-Xi(i-j+k+1))*N(k);
        end
        if k ~= 1
            temp = temp + (xi-Xi(i-j+k))/(Xi(i+k-1)-Xi(i-j+k))*saved;
        end
        saved = N(k);
        N(k) = temp;
    end
    if j == p
        N_tilde = N(1:end-1);
    end
end
if p == 1
    N_tilde = 1;
end

dNdxi = zeros(1,p+1);
dNdxi(1:p)   = -p*N_tilde./(Xi(i+1:i+p)-Xi(i-p+1:i));
dNdxi(2:p+1) = dNdxi(2:p+1) + p*N_tilde./(Xi(i+1:i+p)-Xi(i+1-p:i));
```

Listing E.3: NURBS3DBasis.m

```
function [R, dRdxi, dRdeta, dRdzeta] = NURBS3DBasis(xi, eta, zeta, ...
                                            p, q, r, Xi, Eta, Zeta, weights)
% This routine compute the (p+1)(q+1)(r+1) nonzero NURBS functions
% and corresponding derivatives at (xi, eta, zeta)

% Input
%       (xi,eta,zeta):      evaluation point
%       p,q,r:              NURBS degrees
```

```
%       Xi, Eta, Zeta:      knot vectors
%       weights:            NURBS weights

% Output
%       R, dRdxi, dRdeta, dRdzeta:      array of the (p+1)(q+1)(r+1) NURBS
%                                       functions and its derivativeswhich
%                                       are nonzero at (xi,eta,zeta)


n = length(Xi) - (p+1);
m = length(Eta) - (q+1);
l = length(Zeta) - (r+1);

i1 = findKnotSpan(n, p, xi, Xi);
i2 = findKnotSpan(m, q, eta, Eta);
i3 = findKnotSpan(l, r, zeta, Zeta);

[N, dNdxi] = Bspline_basisDers(i1, xi, p, Xi);
[M, dMdeta] = Bspline_basisDers(i2, eta, q, Eta);
[L, dLdzeta] = Bspline_basisDers(i3, zeta, r, Zeta);

R = zeros(1, (p+1)*(q+1)*(r+1));
dRdxi = zeros(1, (p+1)*(q+1)*(r+1));
dRdeta = zeros(1, (p+1)*(q+1)*(r+1));
dRdzeta = zeros(1, (p+1)*(q+1)*(r+1));

W = 0;
dWdxi = 0;
dWdeta = 0;
dWdzeta = 0;

for k3 = 1:r+1
    A3 = i3 - r + k3 - 1;
    for k2 = 1:q+1
        A2 = i2 - q + k2 - 1;
        for k1 = 1:p+1
            A1 = i1 - p + k1 - 1;
            A = (m*n)*(A3-1) + n*(A2-1) + A1;
            weight = weights(A);

            W       = W       + N(k1)    *M(k2)      *L(k3)       *weight;
            dWdxi   = dWdxi   + dNdxi(k1)*M(k2)      *L(k3)       *weight;
            dWdeta  = dWdeta  + N(k1)    *dMdeta(k2)*L(k3)       *weight;
            dWdzeta = dWdzeta + N(k1)    *M(k2)      *dLdzeta(k3)*weight;
        end
    end
end

counter = 1;
for k3 = 1:r+1
    A3 = i3 - r + k3 - 1;
    for k2 = 1:q+1
        A2 = i2 - q + k2 - 1;
        for k1 = 1:p+1
            A1 = i1 - p + k1 - 1;
            A = (m*n)*(A3-1) + n*(A2-1) + A1;
            fact = weights(A)/(W*W);
```

```
            NML = N(k1)*M(k2)*L(k3);
            R(counter) = NML*fact*W;

            dRdxi(counter)   = (dNdxi(k1)  *M(k2)*L(k3)*W - ...
                NML*dWdxi)*fact;
            dRdeta(counter)  = (dMdeta(k2) *N(k1)*L(k3)*W - ...
                NML*dWdeta)*fact;
            dRdzeta(counter) = (dLdzeta(k3)*N(k1)*M(k2)*W - ...
                NML*dWdzeta)*fact;
            counter = counter + 1;
        end
    end
end
```

Listing E.4: buildGlobalMatrices.m

```
function [K, M, F] = buildGlobalMatrices(varCol, newOptions)
% Create IGA global matrices
% Implemented for linear elasticity operator and the laplace operator, with
% possibility of computing the mass matrix and loading vector from body
% force. Thus, the function handles both static and dynamic linear
% elasticity, laplace- and poisson equation, and dynamic versions of these.

%% Interpret input arguments

% set default values
options = struct('operator','Laplace',...
                 'fieldDimension',1,...
                 'buildMassMatrix',0,...
                 'applyBodyLoading',0);

% read the acceptable names
optionNames = fieldnames(options);

% count arguments
nArgs = length(newOptions);
if round(nArgs/2) ~= nArgs/2
    error('Must have propertyName/propertyValue pairs')
end

for pair = reshape(newOptions,2,[]) %# pair is {propName;propValue}
    inpName = pair{1}; %# make case insensitive

    if any(strcmp(inpName,optionNames))
        options.(inpName) = pair{2};
    else
        error('%s is not a recognized parameter name',inpName)
    end
end

%% Extract all needed data from options and varCol
d = options.fieldDimension;

Xi = varCol.nurbs.knots{1};
Eta = varCol.nurbs.knots{2};
Zeta = varCol.nurbs.knots{3};
p = varCol.nurbs.degree(1);
```

```
q = varCol.nurbs.degree(2);
r = varCol.nurbs.degree(3);

index = varCol.index;
noElems = varCol.noElems;
elRangeXi = varCol.elRangeXi;
elRangeEta = varCol.elRangeEta;
elRangeZeta = varCol.elRangeZeta;
element = varCol.element;
weights = varCol.weights;
controlPts = varCol.controlPts;
runInParallell = varCol.runInParallell;
noCtrlPts = varCol.noCtrlPts;
noDofs = varCol.noDofs;

if strcmp(options.operator,'linearElasticity')
    C = varCol.C;
else
    C = 0; % Will not be used
end

%% Preallocation and initiallizations
n_en = (p+1)*(q+1)*(r+1);

spIdxRow = zeros((d*n_en)^2,noElems);
spIdxCol = zeros((d*n_en)^2,noElems);
Kvalues = zeros((d*n_en)^2,noElems);

if options.buildMassMatrix
    Mvalues = zeros((d*n_en)^2,noElems);
end
if options.applyBodyLoading
    F_indices = zeros(d*n_en,noElems);
    Fvalues = zeros(d*n_en,noElems);
end

[W3D,Q3D] = gaussianQuadNURBS(p+1,q+1,r+1);

%% Build global matrices
parfor (e = 1:noElems, runInParallell)
    idXi = index(e,1);
    idEta = index(e,2);
    idZeta = index(e,3);

    Xi_e = elRangeXi(idXi,:);
    Eta_e = elRangeEta(idEta,:);
    Zeta_e = elRangeZeta(idZeta,:);

    J_2 = ...
        0.125*(Xi_e(2)-Xi_e(1))*(Eta_e(2)-Eta_e(1))*(Zeta_e(2)-Zeta_e(1));

    sctr = element(e,:);
    pts = controlPts(sctr,:);
    sctr_k_e = zeros(1,d*n_en);
    for i = 1:d
        sctr_k_e(1+(i-1)*n_en:i*n_en) = sctr+(i-1)*noCtrlPts;
    end
    k_e = zeros(d*n_en);
```

```
    if options.buildMassMatrix
        m_e = zeros(d*n_en);
    end
    if options.applyBodyLoading
        f_e = zeros(d*n_en,1);
    end

    for gp = 1:size(W3D,1)
        pt = Q3D(gp,:);
        wt = W3D(gp);

        xi   = parent2ParametricSpace(Xi_e,  pt(1));
        eta  = parent2ParametricSpace(Eta_e, pt(2));
        zeta = parent2ParametricSpace(Zeta_e,pt(3));

        [R_fun, dRdxi, dRdeta, dRdzeta] = NURBS3DBasis(xi, eta, zeta, ...
           p, q, r, Xi, Eta, Zeta, weights);

        J = pts'*[dRdxi' dRdeta' dRdzeta'];
        J_1 = det(J);
        dRdX = J'\[dRdxi; dRdeta; dRdzeta];

        switch options.operator
            case 'linearElasticity'
                B = strainDispMatrix3d(n_en,dRdX);
                k_e = k_e + B' * C * B * abs(J_1) * J_2 * wt;
                if options.buildMassMatrix
                    m_e = m_e + blkdiag(R_fun'*R_fun, R_fun'*R_fun, ...
                       R_fun'*R_fun) * abs(J_1) * J_2 * wt;
                end
            case 'laplace'
                k_e = k_e + dRdX'*dRdX* abs(J_1) * J_2 * wt;
                if options.buildMassMatrix
                    m_e = m_e + R_fun'*R_fun * abs(J_1) * J_2 * wt;
                end
        end


        if options.applyBodyLoading
            v = R_fun*pts;
            f_gp = varCol.f(v(1),v(2),v(3));
            f_e = f_e + [f_gp(1)*R_fun'; f_gp(2)*R_fun'; ...
               f_gp(3)*R_fun'] * abs(J_1) * J_2 * wt;
        end
    end

    spIdxRow(:,e) = copyVector(sctr_k_e,d*n_en,1);
    spIdxCol(:,e) = copyVector(sctr_k_e,d*n_en,2);
    Kvalues(:,e) = reshape(k_e, (d*n_en)^2, 1);

    if options.buildMassMatrix
        Mvalues(:,e) = reshape(m_e, (d*n_en)^2, 1);
    end
    if options.applyBodyLoading
        F_indices(:,e) = sctr_k_e';
        Fvalues(:,e) = f_e;
    end
end
```

```
%% Collect data into global matrices (and load vector)
if options.applyBodyLoading
    F = vectorAssembly(Fvalues,F_indices,noDofs);
end

K = sparse(spIdxRow,spIdxCol,Kvalues);
if options.buildMassMatrix
    M = sparse(spIdxRow,spIdxCol,Mvalues);
else
    M = [];
end

if min(size(K)) < noDofs
    K(noDofs,noDofs) = 0;
    if options.buildMassMatrix
        M(noDofs,noDofs) = 0;
    end
end
```

Bibliography

[1] Thomas JR Hughes, John A Cottrell, and Yuri Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39):4135–4195, 2005.

[2] J. Austin Cottrell, Thomas J. R. Hughes, and Yuri Bazilevs. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley Publishing, New Jersey, USA, 1st edition, 2009.

[3] F. Ihlenburg. *Finite Element Analysis of Acoustic Scattering*. Applied Mathematical Sciences. Springer, New York, USA, 1998.

[4] Stéphane G Conti, Philippe Roux, Christian Fauvel, Benjamin D Maurer, and David A Demer. Acoustical monitoring of fish density, behavior, and growth rate in a tank. *Aquaculture*, 251(2):314–323, 2006.

[5] Dang Manh Nguyen, Jens Gravesen, and Anton Evgrafov. *Isogeometric analysis and shape optimization in electromagnetism*. PhD thesis, Technical University of DenmarkDanmarks Tekniske Universitet, Department of Informatics and Mathematical ModelingInstitut for Informatik og Matematisk Modellering, 2012.

[6] Nguyen Dang Manh, Anton Evgrafov, Allan Roulund Gersborg, and Jens Gravesen. Isogeometric shape optimization of vibrating membranes. *Computer Methods in Applied Mechanics and Engineering*, 200(13):1343–1353, 2011.

[7] L Beirao da Veiga, A Buffa, J Rivas, and G Sangalli. Some estimates for h–p–k-refinement in Isogeometric Analysis. *Numerische Mathematik*, 118(2):271–305, 2011.

[8] L Beirao da Veiga, A Buffa, G Sangalli, and R Vázquez. Mathematical analysis of variational isogeometric methods. *Acta Numerica*, 23:157–287, 2014.

[9] Peter Nørtoft, Jens Gravesen, and Morten Willatzen. Isogeometric analysis of sound propagation through laminar flow in 2-dimensional ducts. *Computer Methods in Applied Mechanics and Engineering*, 284:1098–1119, 2015.

[10] David S Burnett. A three-dimensional acoustic infinite element based on a prolate spheroidal multipole expansion. *The Journal of the Acoustical Society of America*, 96(5):2798–2816, 1994.

[11] Klaus Gerdes and L Demkowicz. Solution of 3D-Laplace and Helmholtz equations in exterior domains using hp-infinite elements. *Computer Methods in Applied Mechanics and Engineering*, 137(3):239–273, 1996.

[12] Bodo Nolte, Ingo Schäfer, Christ de Jong, and Lauton Gilroy. BeTSSi II Benchmark on Target Strength Simulation. 2014. http://www.fa2014.agh.edu.pl/fa2014_cd/article/SS/SS17_5.pdf.

[13] I.A. Stegun. *Handbook of Mathematical Function: With Formulas Graphs, and Mathematical Tables*. Dover, New York, USA, 1965.

[14] Y.C. Chang. *Vibrations of a spherical shell comparison of 3-D elasticity and Kirchhoff shell theory results*. Number no. 6 in TICAM report. Texas Institute for Computational and Applied Mathematics, University of Texas at Austin, 1994.

[15] Tom Lyche and Knut Morken. Spline methods draft. *Department of Informatics, University of Oslo*, 2004. http://heim.ifi.uio.no/knutm/komp04.pdf.

[16] Qi-Xing Huang, Shi-Min Hu, and Ralph R Martin. Efficient Degree Elevation and Knot Insertion for B-spline Curves using Derivatives. *Computer-Aided Design and Applications*, 1(1-4):719–725, 2004.

[17] Vinh Phu Nguyen, Stéphane Bordas, and Timon Rabczuk. Isogeometric analysis: an overview and computer implementation aspects. *arXiv preprint arXiv:1205.2129*, 2012.

[18] D. Ramakrishnan. D. Ramakrishnan's Course Notes: Chapter 8: Change of Variables, Parametrizations, Surface Integrals. pages 7 – 8. http://www.math.caltech.edu/~dinakar/dr_notes.html.

[19] Siv Bente Raknes. Isogeometric Analysis and Degenerated Mappings. Master's thesis, Trondheim, Norway, 2011. http://www.diva-portal.org/smash/get/diva2:415385/FULLTEXT01.pdf.

[20] Kjetil André Johannessen, Trond Kvamsdal, and Tor Dokken. Isogeometric analysis using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, 269:471–514, 2014.

[21] Yuri Bazilevs, Victor M Calo, John A Cottrell, John A Evans, TJR Hughes, S Lipton, MA Scott, and TW Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5):229–263, 2010.

[22] Tor Dokken, Tom Lyche, and Kjell Fredrik Pettersen. Polynomial splines over locally refined box-partitions. *Computer Aided Geometric Design*, 30(3):331–356, 2013.

[23] Kjetil André Johannessen, Mukesh Kumar, and Trond Kvamsdal. Divergence-conforming discretization for Stokes problem on locally refined meshes using LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, 2015.

[24] Kjetil André Johannessen, Filippo Remonato, and Trond Kvamsdal. On the similarities and differences between Classical Hierarchical, Truncated Hierarchical and LR B-splines. *Computer Methods in Applied Mechanics and Engineering*, 291:64–101, 2015.

[25] Thomas JR Hughes, Alessandro Reali, and Giancarlo Sangalli. Isogeometric methods in structural dynamics and wave propagation. *Proceedings of COMPDYN 2009-Computational Methods in Structural Dynamics and Earthquake Engineering*, pages 169–186, 2009.

[26] JA Cottrell, A Reali, Y Bazilevs, and TJR Hughes. Isogeometric analysis of structural vibrations. *Computer methods in applied mechanics and engineering*, 195(41):5257–5296, 2006.

[27] Leonard Meirovitch. *Analytical methods in vibrations*. Macmillan series in advanced mathematics and theoretical physics. Macmillan, New York, USA, 1967.

[28] R Schmidt, J Kiendl, K-U Bletzinger, and R Wüchner. Realization of an integrated structural design process: analysis-suitable geometric modelling and isogeometric analysis. *Computing and visualization in science*, 13(7):315–330, 2010.

[29] Jean-Pierre Berenger. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics*, 114(2):185–200, 1994.

[30] J.-P. Berenger. Perfectly matched layer for the FDTD solution of wave-structure interaction problems. *Antennas and Propagation, IEEE Transactions on*, 44(1):110–117, Jan 1996.

[31] Calvin H Wilcox. An expansion theorem for electromagnetic fields. *Communications on Pure and Applied Mathematics*, 9(2):115–134, 1956.

[32] H. Triebel. Leis, R., Initial Boundary Value Problems in Mathematical Physics. Chichester etc., John Wiley & Sons; Stuttgart, B. G. Teubner 1986. VIII, 266 pp., DM 62,—. ISBN 3-519-02102-1. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 68(12):654–654, 1988.

[33] K Gerdes. The conjugated vs. the unconjugated infinite element method for the Helmholtz equation in exterior domains. *Computer Methods in Applied Mechanics and Engineering*, 152(1):125–145, 1998.

[34] Ivo M. Babuska and Stefan A. Sauter. Is the Pollution Effect of the FEM Avoidable for the Helmholtz Equation Considering High Wave Numbers? *SIAM Review*, 42(3):451–484, 2000.

[35] Alexander Karatarakis, Panagiotis Karakitsios, and Manolis Papadrakakis. Computation of the isogeometric analysis stiffness matrix on GPU. In *Proc. of the 3rd South-East European Conference on Computational Mechanics (SEECCM)*, 2013.

[36] Thomas JR Hughes, A Reali, and G Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5):301–313, 2010.

[37] Niclas Jansson. Optimizing Sparse Matrix Assembly in Finite Element Solvers with One-Sided Communication. In *High Performance Computing for Computational Science-VECPAR 2012*, pages 128–139. Springer, 2013.

[38] Timothy A Davis. Algorithm 930: FACTORIZE: an object-oriented linear system solver for MATLAB. *ACM Transactions on Mathematical Software (TOMS)*, 39(4):28, 2013.

[39] Markus Clemens and Thomas Weiland. Iterative methods for the solution of very large complex symmetric linear systems of equations in electrodynamics. *Technische Hochschule Darmstadt, Fachbereich*, 18, 2002.

[40] Andreas Asheim. *Numerical methods for highly oscillatory problems.* PhD thesis, Trondheim, Norway, 2010.

[41] Christopher W Nell and Layton E Gilroy. An improved BASIS model for the BeTSSi submarine. *DRDC Atlantic TR*, 199:2003, 2003.

[42] Hans G Schneider, R Berg, L Gilroy, I Karasalo, I MacGillivray, MT Morshuizen, and A Volker. Acoustic scattering by a submarine: results from a benchmark target strength simulation workshop. *ICSV10*, pages 2475–2482, 2003.

[43] Ralf Burgschweiger, Ingo Schäfer, Martin Ochmann, and Bodo Nolte. Results of the ray-tracing based solver BEAM for the approximate determination of acoustic backscattering from thin-walled objects. In *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, volume 249, pages 1218–1227. Institute of Noise Control Engineering, 2014.

[44] Phillip L Gould. *Introduction to linear elasticity.* Springer, New York, USA, 1994.

[45] William S Slaughter. *The linearized theory of elasticity.* Springer Science & Business Media, Boston, USA, 2002.

[46] Ted Belytschko and Itai Leviathan. Physical stabilization of the 4-node shell element with one point quadrature. *Computer Methods in Applied Mechanics and Engineering*, 113(3):321–350, 1994.