Analytical and semi-analytical methods for buckling and bending of steel elastomer sandwich plates

by

OLE J. HAREIDE

THESIS

for the degree of

MASTER OF SCIENCE

(Master i Anvendt matematikk og mekanikk)

Faculty of Mathematics and Natural Sciences
University of Oslo

May 2012

Matematisk-naturvitenskaplig fakultet
Universitetet i Oslo

Preface

This thesis has been written for the submission to the Master of Science (MSc) degree at the University of Oslo (UiO), Faculty of Mathematics and Natural Sciences, Department of Mathematics. The work of the thesis has been carried out between January 2011 and May 2012.

The work has been done in collaboration with the Ship Structures and Concepts department (NTANO367) of the Maritime Advisory Division at Det Norske Veritas (DNV). During the course of the thesis I have been fortunate to be given a desk at DNV's head office at Høvik, and all the analyses and the writing of the thesis has been done there.

I would like to thank my two primary supervisors, Dr. Lars Brubak and Dr. Scient. Eivind Steen, at NTANO367 for their excellent support and guidance throughout these months. I would also like to express my gratitude to Professor Jostein Hellesland at the Mechanics Division, Department of Mathematics, Faculty of Mathematics and Natural Sciences, University of Oslo for his guidance and advice. My fellow students Henning Levanger and Mathias Sørby Haugen, both students at UiO and writing theses at DNV, also deserve acknowledgement for the many discussions and the inspiration they have brought to our small group.

Finally, I would like to dedicate the work of this thesis to the memory of my mother, Bjørg Beston Hareide, who was diagnosed with terminal cancer in April 2011 and died 4 September 2011.

Oslo, May 2012.

Ole J. Hareide

Abstract

This thesis has been written about the analytical and semi-analytical formulae as an alternative to finite element analyses in computing lateral deflection, buckling loads and ultimate capacity of steel elastomer sandwich plates. Since the introduction of sandwich plates in the aerospace industry several decades ago, the shipbuilding industry is now finding a use of their own for this type of construction. The major classification societies are all in the process of creating rules for the design of sandwich plates for use in the maritime industry and in this process Det Norske Veritas has created the classification note CN.30.11 Steel Sandwich Panel Construction [1].

The thought behind these rules is to use formulae in the design of the sandwich panels instead of the more time consuming finite element method. The work in this thesis has involved finding the proper analytical formulae for buckling and bending of simply supported sandwich plates from relevant theory and comparing the results from these formulae with the finite element method. The formulae have been implemented in a Visual Basic-automated Excel spread-sheet which has been made available as a supporting tool for the classification note.

Another part of the thesis has been to develop a semi-analytical method, using a Rayleigh-Ritz approach, for more accurate buckling strength assessments. This model can be used for both simply supported and clamped plates and calculates eigenvalues for all combinations of in-plane normal and shear forces. It also can accurately account for pre-stresses in the buckling analyses.

The results from the analyses are presented along with the underlying theory in this thesis. For the analytical models the results show a good agreement between the formulae and the finite element method for most cases, with the closed form formulae often being a bit conservative on the safe side. The results for the Rayleigh-Ritz model also show very good agreement with the finite element method. Some problems were encountered in the modelling procedures of sandwich plates in ABAQUS, especially with the solid element models. This can be the cause of some of the difference between the analytical and finite element models in the results.

Contents

List of notations

$\bar{\lambda}$	Plate reduced slenderness
$\boldsymbol{\Phi}$	Vector with surface forces
$\boldsymbol{\sigma}$	Vector with stress components
$\boldsymbol{\varepsilon}$	Vector with strain components
$\boldsymbol{A}$	Vector with generalized degrees of freedom
$\boldsymbol{E}$	Material matrix
$\boldsymbol{F}$	Vector with body forces
$\boldsymbol{K}^G$	Geometric stiffness matrix
$\boldsymbol{K}^M$	Material stiffness matrix
$\boldsymbol{K}^S$	Elastic spring stiffness matrix
Δ	Laplace operator
δ	Variational operator
δ_{ij}	Kronecker's delta
Λ	Load factor
Λ_B	Load factor for elasto-plastic buckling
Λ_E	Load factor at elastic buckling - uniaxial loading
Λ_F	Load factor for face plate material yield
$\Lambda_{E,\tau}$	Load factor at elastic buckling - shear loading
$\Lambda_{E,bi+\tau}$	Load factor at elastic buckling - combined biaxial and shear loading

$\Lambda_{E,bi}$	Load factor at elastic buckling - biaxial loading
ν_c	Poisson's number for core layer
ν_{f1}	Poisson's number for upper face layer
ν_{f2}	Poisson's number for lower face layer
ϕ_x	Actual cross-sectional rotation in the x-direction (about the y-axis)
ϕ_y	Actual cross-sectional rotation in the y-direction (about the x-axis)
Π	Total potential energy
σ_F	Face plate yield stress
σ_{10}	Face plate normal stress in x-direction
σ_{20}	Face plate normal stress in y-direction
τ_0	Face plate shear stress
θ	Shear factor
$\tilde{\mathbf{K}}^G$	Condensed geometric stiffness matrix
$\tilde{\mathbf{K}}^M$	Condensed material stiffness matrix
a	Dimension of plate in x-direction
a_i	Degrees of freedom
A_{ij}, B_{ij}, C_{ij}	Amplitude coefficients in Rayleigh-Ritz series
b	Dimension of plate in y-direction
D	Flexural rigidity for isotropic sandwich plate
d	Distance between centroids of face layers
D_{11}	Flexural rigidity in x-direction
D_{22}	Flexural rigidity in y-direction
D_{33}	Flexural rigidity in xy-direction
E_c	Young's modulus for core layer

E_{f1}	Young's modulus for upper face layer
E_{f2}	Young's modulus for lower face layer
H	Potential of external loads
K	Buckling coefficient
k	Spring stiffness
K_0	Buckling coefficient
m	Number of terms in series, number of half waves in x-direction
M, N, P, Q	Maximum number of terms (degrees of freedom) in Rayleigh-Ritz series
n	Number of terms in series, number of half waves in y-direction
P_x	Compressive in-plane normal force intensity in x-direction
P_y	Compressive in-plane normal force intensity in y-direction
P_{xy}	Compressive in-plane shear force intensity
q_{mn}	Load amplitude coefficient in Fourier series
S	Shear stiffness for isotropic sandwich plate
S_x	Shear stiffness in x-direction
S_y	Shear stiffness in x-direction
t_c	Core layer thickness
t_{f1}	Upper face layer thickness
t_{f2}	Lower face layer thickness
t_{sps}	Total sandwich plate thickness
U	Internal strain energy
u	Displacement in x-direction
v	Displacement in y-direction
w	Displacement in z-direction

w_b	Part of total deflection in z-direction due to bending deformations
w_s	Part of total deflection in z-direction due to transverse shear deformations
W_{ext}	External work
W_{int}	Internal work
w_{mn}	Displacement amplitude coefficient in Fourier series
x, y, z	Spatial co-ordinates

Chapter 1

Introduction

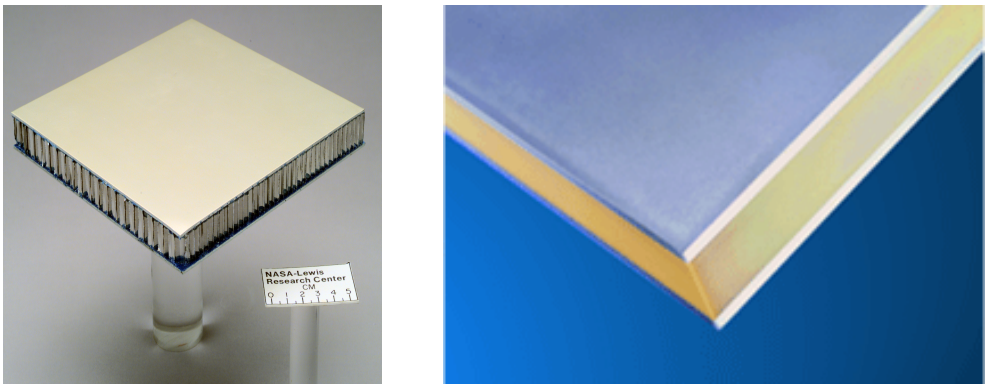
1.1 Background

With the introduction of sandwich panels in shipbuilding there are several advantages to be gained. Increased stiffness to weight ratio and a geometrically simpler structure may be the two most important properties of this type of panels over traditionally stiffened steel plates. One type of such a panel is the Sandwich Plate System (SPS)[1]. With this new technology Det Norske Veritas (DNV) is currently working on new guidelines for the SPS in order to correctly design such panels for use in ships. In this ongoing project, there has already been one master thesis written about SPS in collaboration between the University of Oslo and DNV, Fladby [2], focusing on the ultimate capacity of these plates subjected to a variety of load combinations.

1.2 Sandwich composite constructions

A sandwich composite is a form of laminated composite made up of three distinct layers of different materials; a thick, lightweight and relatively low-performing core bonded on either side by a thin, stiff and relatively high-performing face plate. Common core materials are honeycomb and corrugated profiles made of metals, glass fibres or even paper and solid cores made from types of light-weight wood or cellular plastic foams. In the same manner the face sheets can also be made from a bewildering range of materials such as laminates of different fibre reinforced composites or different types of metals and wood. In short, any material can be used as a face material provided it can be produced in a thin sheet form. This apparent freedom to choose among so many different materials is one of the major

[1]The SPS is a plated system patented by Intelligent Engineering, www.ie-sps.com



(a) **(b)**

Figure 1.1: (a) Glass-fibre reinforced aluminium faces and honeycomb core typically used in aerospace industry (image courtesy of NASA) and (b) steel faces and elastomer core (SPS-panel) typically used in heavy engineering (image courtesy of Intelligent Engineering)

advantages of sandwich composites and makes it possible to tailor the structure to different applications [3]. Two different types of sandwich panels are shown in Figs. 1.1a and 1.1b.

The other major advantage of sandwich constructions is the ratio of stiffness over weight. Laminates of fibre reinforced composites can be made with strength matching that of metals but with stiffness often several magnitudes lower. In placing the high-performing materials farther away from the neutral plane by separating them with a lightweight core, the face sheets become analogous with the flanges of an I-beam, a highly optimized structure which leaves just enough material in the web to keep the flanges separated. The core will thus play the same role in separating the faces and transferring shear stresses, but in sandwich structures (especially in plates) it can also play secondary roles such as contributing to damping in oscillatory motions, absorption of impact energy from collisions or blast waves, soundproofing and fire and thermal insulation. In this, the adhesive used to bond the face sheets to the core is perhaps as important to the sandwich as the material layers. If the faces and core are not properly bonded together they will not perform as a single integrated structure and will not achieve the advantageous properties mentioned above.

The history of sandwich can be summed up as follows. The first proper use of a sandwich concept in a construction was perhaps Robert Stephenson's locomotive

Planet built in 1830. By cladding a wooden beam with wrought iron plates bolted onto the wood, he utilized the properties of two different materials, thereby saving weight and maintaining the required performance. Just after the turn of the last century, different honeycomb cores appeared, not for structural applications at first, but for bee keeping and decorative paper arts. Several other honeycomb cores were developed but it was not until the late 1930s that an adhesive was developed that made honeycomb cores possible in structural applications. During the Second World War appears the British Mosquito combat aircraft made from balsa core and veneer faces, a consequence of war-time shortage of the metals normally used in aircrafts. After the war the aerospace industry embraced the honeycomb cores, which to this day still offers the highest shear stiffness to weight ratio, but the high cost of this type of sandwich prevails and it has not found any notable use outside the aerospace industry. In the late 1950s and early 1960s the cellular foams invented during World War II were finally ready for use in structural application, and in contrast with the complex honeycomb cores, the foams have found widespread use in sandwich panels in low- and medium-cost applications. Few new core materials have appeared since then, research has mainly focused on the bonding of core and face materials and different face materials [3].

1.3 SPS - Sandwich Plate System

There is large focus in the maritime and offshore industries to come up with solutions which will yield lighter structures, better fuel economy and simpler maintenance and repair. Because of this, several new innovations appear. It is not possible to develop specific design guidelines for all the various solutions, but some products have indeed reached a certain level in both market position and product development that they merit further attention. One such system is the Sandwich Plate System, or SPS.

SPS is a patented sandwich plate system made by Intelligent Engineering in Canada. Made up of two metal face sheets bonded to a polyurethane elastomer core, it has since its introduction in the mid-1990s been installed in new ships and offshore structures, terraces in stadia and arenas, bridge decks, structural floors in buildings and used in repairs of structural systems in ship and offshore structures. The biggest advantage associated with SPS panels is perhaps the increased stiffness to weigh ratio which makes elimination of the lowest level of stiffeners in a conventional plated structures possible. This also leads to a less complicated structure with fewer welds, fewer surfaces to be coated and easier maintenance of

the structures in use. Other benefits associated with SPS include prefabrication of elements, which ensures simpler installation procedures and shortens the construction time, and the ease with which the system can be used in repair of damaged ship components. Recently major shipyards in Asia have secured contracts with Intelligent Engineering to mass produce ship components using SPS-technology.

1.4 Objective and scope of this thesis

This thesis will focus mainly on the verification of models for elastic buckling and lateral pressure deformations, in addition to some simplified elasto-plastic models for conservative ultimate strength assessments. The models to be verified include both analytically closed-form formulae and Rayleigh-Ritz semi-analytical models. The analytical models will include a lateral pressure deflection model and a buckling of combined in-plane normal and shear loading for simply supported plates. The semi-analytical model will exclude bending deflection analyses, but will for buckling also include elastic springs on the boundary, thereby being able to model both simply supported and clamped plates, and plates with pre-stresses. The theories in this thesis all assume small deformation theory and thus the post-critical reserve of plates is not considered, neither are the effects such as wrinkling of the individual face sheets, shear fracture of the core and de-bonding of the face/core interface.

A small convergence study has been done in ABAQUS before the verification models were made. The analytical theories have been implemented in an automated Excel spreadsheet which has now been supplied as a support tool along with the DNV classification note [1]. This tool has also been used to verify the analytical formulae for this thesis. In addition to that, the Rayleigh-Ritz model has been implemented in FORTRAN as an alternative to the analytical formulae and the Finite Element method. The results from the verifications are presented in graphs in this thesis and the results are discussed.

1.5 Presentation of chapters

In Chapter 1 a short introduction into the general history and theory of sandwich structures, and especially plates, is presented. A short presentation of the scope

of this thesis and a description of the chapters are also found here.

In Chapter 2 the analytical formulae for bending and buckling of simply supported sandwich plates are derived. The thin plate theory for isotropic plates is extended by including the effect on transverse shear.

In Chapter 3 an overview of potential energy in plates and two approximate methods for finding the eigenvalues in buckling analyses are presented, the Rayleigh-Ritz method and the Finite Element method.

In Chapter 4 the Rayleigh-Ritz model which was made during this thesis is presented. It includes both simply supported and clamped plates with or without pre-stresses.

In Chapter 5 a summary of the modelling procedure in the Finite Element software ABAQUS is presented along with a small convergence study on which elements were to be used, and a summary on some of the most common problems encountered during the modelling procedures.

In Chapter 6 the automated Excel spreadsheet SPS-calculator is presented. It is an implementation of the formulae presented in Chapter 2, and has been developed simultaneously with this thesis and the work on the classification note CN.30.11[1].

In Chapter 7 the results from verification of the analytical and semi-analytical (Rayleigh-Ritz) models is presented in graphical form and discussed.

In Chapter 8 the conclusions of the work carried out in this thesis are presented along with a suggestion on further work.

Chapter 2

Sandwich plate theory

2.1 Introduction

In plates with length to thickness ratio approaching 10 or lower [4], or plates with materials of low shear stiffness, the classical plate theory (CPT) found in literature like Timoshenko [5], Brush and Almroth [6] or Reddy [4] will no longer not yield accurate solutions to typical plate problems like deflections, buckling and vibrations. The reason is that this theory assumes that the cross-sectional rotations stem from bending curvatures only, whereas in the sandwich plates described in this thesis, and indeed in thick isotropic steel plates, the cross-sectional rotations will arise from both curvatures and shear deformations which are induced by either low shear stiffness, high shear stresses through the thickness of the plate or both. This makes the CPT predict a stiffer plate than shear deformation theories, giving erroneous deflections and eigenvalues.

2.2 Displacement field

In deriving the classical plate theory (CPT), or thin plate theory, a set of simplifications known as the *Kirchhoff's assumptions* are fundamental. These appear in different formulations in the literature, in Reddy [4] they are given as:

1. Straight lines perpendicular to the mid-surface before deformations remain straight after deformation.

2. The transverse normals do not experience elongation - they are inextensible.

3. The transverse normals rotate such that they remain perpendicular to the middle plane after deformation.

In sandwich plates, the comparatively low transverse shear stiffness of the core can have a profound impact on the performance of the plate, and the effects due to transverse and in-plane shearing of the core layer cannot be correctly described by the classical thin plate theory. By extending the classical theory to also include these deformations, sandwich plates can be described to a far greater accuracy. This theory is known as Mindlin-Reissner theory and was originally developed for thick isotropic plates. Also known as first-order shear deformation theory, it relaxes the third Kirchhoff assumption, allowing the cross-sectional normals to rotate such that they remain straight but no longer normal to the deformed middle plane. A further relaxation of the straightness assumption, number one in the list above, is the basis for a third-order shear deformation theory, where the initially straight normals may deform into cubic curves over the cross sections. For more details on third order shear deformation theories, see Reddy [4]. For the type of sandwich plates studied in this thesis, isotropic steel-elastomer sandwich, only first-order shear deformations are relevant and included in the theory presented here.

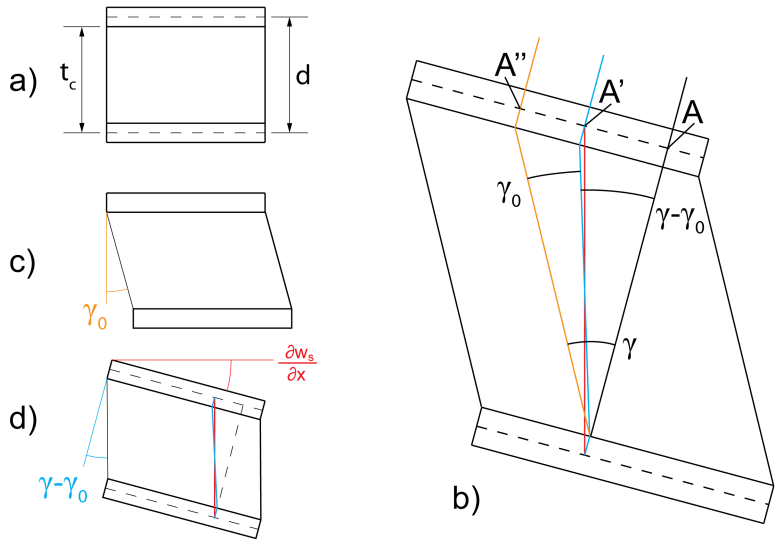


Figure 2.1: Breakdown of shear deformations in a sandwich plate. The undeformed sandwich element shown in (a) is deformed as shown in (b) from both in-plane (c) and transverse (d) shear strains.

Fig. 2.1 shows a small element of a sandwich plate subjected to shear deformations. The total shear deformations, γ, is the sum of in-plane and out-of-plane (transverse) shear deformations, as shown in the figure. The out-of-plane shear leads to a sliding of the cross-sections with respect to each other, while the in-plane shear induces a sliding of the face sheets relative to each other. In both cases it

is assumed that the shear deformations happen in the core only, and that the face plates are stiff in shear compared to the core. The red line in the figure represents shear over the whole cross-section, while the blue line assumes shear deformations in the core only. The distance A to A' is, however, the same and shows the horizontal relative sliding of the face sheets coming from the out-of-plane shear. The angle between the blue and the orange line is the in-plane shear γ_0 and the distance A' to A'' shows the corresponding horizontal sliding.

The displacement of a point in the plate is (u, v, w) and is defined as:

$$\begin{aligned} u(x, y, z) &= u_0(x, y) + z\phi_x(x, y) \\ v(x, y, z) &= v_0(x, y) + z\phi_y(x, y) \\ w(x, y, z) &= w_0(x, y) \end{aligned} \tag{2.2.1}$$

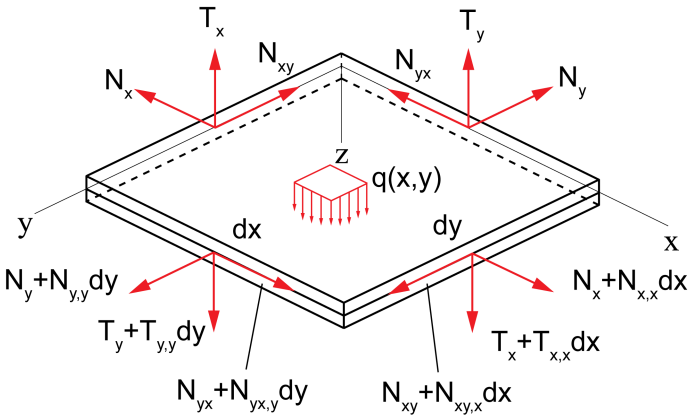
The main difference from classical plate theory is that the actual cross-sectional rotations, ϕ_x and ϕ_y, are now not only due to the bending of the middle plane but also the transverse shear deformations. They are found by differentiating the displacement field in Eq. (2.2.1):

$$\phi_x = \frac{\partial u}{\partial z} \qquad \text{and} \qquad \phi_y = \frac{\partial v}{\partial z} \tag{2.2.2}$$

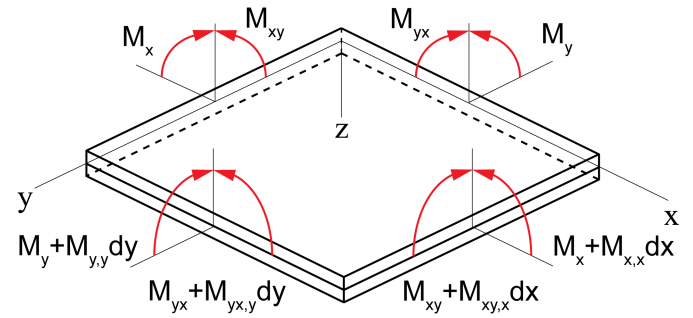
The displacement field in Eq. (2.2.1) is from Reddy [4] and is a simple extension of classical plate theory to include transverse shear deformations. This field is probably best used to describe isotropic plates which can experience these deformations, i.e. thick plates where the length over thickness ratio is approaching 10 and below, and assumes that the entire cross-sections experience the same constant rotations. In sandwich plates the core is assumed to experience all the shear deformations, and the faces should therefore not undergo the same rotations as the core. The same displacement field is nonetheless used in the sandwich plate theory presented by Zenkert [3] in what is called thin-face approximation.

2.3 Equilibrium equations

The equations of equilibrium for the plate are derived from an infinitesimal element of size $dx\ dy$ shown in Fig. 2.2. For graphic simplicity the element is first shown with in-plane and transverse force intensities in one figure and with bending and twisting moment intensities in another figure. To get the total force acting on a side of the element, the force intensity is multiplied by the length of the side, either



(a)



(b)

Figure 2.2: Transverse and in-plane force intensities (a) and bending and twisting moment intensities (b) acting on a small element of the plate

dx or dy. Summation of forces in the x-direction gives:

$$-N_x dy + \left(N_x + \frac{\partial N_x}{\partial x} dx\right) dy - N_{yx} dx + \left(N_{yx} + \frac{\partial N_{yx}}{\partial y} dy\right) dx = 0$$

$$\Rightarrow \frac{\partial N_x}{\partial x} + \frac{\partial N_{yx}}{\partial y} = 0 \qquad (2.3.1)$$

and similar summation of forces in the y-direction gives:

$$-N_y dx + \left(N_y + \frac{\partial N_y}{\partial y} dy\right) dx - N_{xy} dy + \left(N_{xy} + \frac{\partial N_{xy}}{\partial x} dx\right) dy = 0$$

$$\Rightarrow \frac{\partial N_y}{\partial y} + \frac{\partial N_{xy}}{\partial x} = 0 \qquad (2.3.2)$$

The summation of forces in the z-direction is, however, not as straight forward as the in-plane forces. As the plate bends, the in-plane forces are assumed to follow the middle plane of the plate, and thus vertical components of the in-plane forces develop. The vertical equilibrium equation must hence be derived from the plate in a slightly bent configuration in order to include this effect.

Beginning with the projection of the N_x force intensity onto the z-axis the contribution from N_x to the vertical equilibrium is found:

$$\begin{aligned} -N_x dy \frac{\partial w}{\partial x} + \left(N_x + \frac{\partial N_x}{\partial x} dx\right) dy \left(\frac{\partial w}{\partial x} + \frac{\partial^2 w}{\partial x^2} dx\right) = \\ -N_x \frac{\partial w}{\partial x} dy + N_x \frac{\partial w}{\partial x} dy + N_x dy \frac{\partial^2 w}{\partial x^2} dx + \frac{\partial N_x}{\partial x} \frac{\partial w}{\partial x} dx\, dy + \frac{\partial N_x}{\partial x} \frac{\partial^2 w}{\partial x^2} dx^2\, dy = \\ N x \frac{\partial^2 w}{\partial x^2} dx\, dy + \frac{\partial N_x}{\partial x} \frac{\partial w}{\partial x} dx\, dy \end{aligned} \tag{2.3.3}$$

where $dx^2 << dx$ and equal terms cancel each other out. Similarly the projection of the N_y force intensity onto the z-axis yield:

$$N_y \frac{\partial^2 w}{\partial y^2} dx\, dy + \frac{\partial N_y}{\partial y} \frac{\partial w}{\partial y} dx\, dy \tag{2.3.4}$$

The contribution from the shearing force intensity N_{xy} is found in the same way:

$$\begin{aligned} -N_{xy} \frac{\partial w}{\partial y} dy + \left(N_{xy} + \frac{\partial N_{xy}}{\partial x} dx\right) dy \left(\frac{\partial w}{\partial y} + \frac{\partial^2 w}{\partial x\, \partial y} dx\right) = \\ -N_{xy} \frac{\partial w}{\partial y} dy + N_{xy} \frac{\partial w}{\partial y} dy + N_{xy} \frac{\partial^2 w}{\partial x\, \partial y} dx\, dy + \frac{\partial N_{xy}}{\partial x} \frac{\partial w}{\partial y} dx\, dy + \frac{\partial N_{xy}}{\partial x} \frac{\partial^2 w}{\partial x\, \partial y} dx^2 dy = \\ N_{xy} \frac{\partial^2 w}{\partial x\, \partial y} dx\, dy + \frac{\partial N_{xy}}{\partial x} \frac{\partial w}{\partial y} dx\, dy \end{aligned} \tag{2.3.5}$$

and similarly for the shearing force intensity N_{yx}:

$$N_{yx} \frac{\partial^2 w}{\partial y\, \partial x} dx\, dy + \frac{\partial N_{yx}}{\partial y} \frac{\partial w}{\partial x} dx\, dy \tag{2.3.6}$$

The final expression for the contribution to the vertical equilibrium is found by summation of vertical components of N_x, N_y, N_{xy} and N_{yx}:

$$\begin{aligned} N_x \frac{\partial^2 w}{\partial x^2} dx\, dy + \frac{\partial N_x}{\partial x} \frac{\partial w}{\partial x} dx\, dy + N_y \frac{\partial^2 w}{\partial y^2} dx\, dy + \frac{\partial N_y}{\partial y} \frac{\partial w}{\partial y} dx\, dy + N_{xy} \frac{\partial^2 w}{\partial x\, \partial y} dx\, dy \\ + \frac{\partial N_{xy}}{\partial x} \frac{\partial w}{\partial y} dx\, dy + N_{yx} \frac{\partial^2 w}{\partial y\, \partial x} dx\, dy + \frac{\partial N_{yx}}{\partial y} \frac{\partial w}{\partial x} dx\, dy \end{aligned}$$

By using the in-plane equations (2.3.1) and (2.3.2) and crossing out the product $dx\,dy$ which appears in all terms, this expression can be reduced to:

$$N_x\frac{\partial^2 w}{\partial x^2} + N_y\frac{\partial^2 w}{\partial y^2} + N_{xy}\frac{\partial^2 w}{\partial x\,\partial y} + N_{yx}\frac{\partial^2 w}{\partial y\,\partial x} \tag{2.3.7}$$

which is the final expression for the contribution of the in-plane normal and shearing forces to the vertical equilibrium. This can now be added to the summation of transverse forces and the pressure, to finally obtain the vertical equilibrium equation:

$$\frac{\partial T_x}{\partial x} + \frac{\partial T_y}{\partial y} + q(x,y) + N_x\frac{\partial^2 w}{\partial x^2} + 2N_{xy}\frac{\partial^2 w}{\partial x\,\partial y} + N_y\frac{\partial^2 w}{\partial y^2} = 0 \tag{2.3.8}$$

As with the equilibrium of forces, one can also derive moment equilibrium equations. The bending and twisting moments acting on the same plate element $dx\,dy$ are shown in Fig. 2.2. The moment equilibrium about the y-axis is:

$$\begin{aligned} -M_x dy + \left(M_x + \frac{\partial M_x}{\partial x}dx\right)dy - M_{yx}dx + \left(M_{yx} + \frac{\partial M_{yx}}{\partial y}dy\right)dx \\ +T_y\frac{dx^2}{2} - \left(T_y + \frac{\partial T_y}{\partial y}dy\right)\frac{dx^2}{2} - \left(T_x + \frac{\partial T_x}{\partial x}dx\right)dy\,dx = \\ T_x - \frac{\partial M_x}{\partial x} - \frac{\partial M_{yx}}{\partial y} = 0 \end{aligned} \tag{2.3.9}$$

Similarly for the moments about the x-axis:

$$T_y - \frac{\partial M_y}{\partial y} - \frac{\partial M_{xy}}{\partial x} = 0 \tag{2.3.10}$$

and taking the moment about the z-axis reveals the known relationship:

$$M_{xy} = M_{yx} \tag{2.3.11}$$

It is quite clear from the derivations of the equations of equilibrium that these are identical for isotropic plates and sandwich plates [3], [5].

By differentiating once and substituting the moment equilibrium equations (2.3.9) and (2.3.10) for the derivatives of transverse forces in the vertical equilibrium equation (2.3.8), one arrives at the differential equations:

$$\frac{\partial^2 M_x}{\partial x^2} + 2\frac{\partial^2 M_{xy}}{\partial x\,\partial y} + \frac{\partial^2 M_y}{\partial y^2} = \frac{\partial T_x}{\partial x} + \frac{\partial T_y}{\partial y} \tag{2.3.12}$$

$$\frac{\partial T_x}{\partial x} + \frac{\partial T_y}{\partial y} = -\left(q + N_x \frac{\partial^2 w}{\partial x^2} + 2N_{xy}\frac{\partial^2 w}{\partial x\,\partial y} + N_y \frac{\partial^2 w}{\partial y^2}\right) \tag{2.3.13}$$

Before introducing the moment-curvature relations to obtain the fourth-order differential equation in terms of the deflection, w, one should consider these relations. Because the sandwich plate has a thick core with low shear stiffness, the transverse shear forces T_x and T_y will also give significant contributions to the total deflection, and hence the proper moment-curvature relations should be:

$$\begin{aligned}
M_x &= -D_{11}\left[\frac{\partial}{\partial x}\left(\frac{\partial w}{\partial x} - \frac{T_x}{S_x}\right) + \nu_{yx}\frac{\partial}{\partial y}\left(\frac{\partial w}{\partial y} - \frac{T_y}{S_y}\right)\right] \\
M_y &= -D_{22}\left[\frac{\partial}{\partial y}\left(\frac{\partial w}{\partial y} - \frac{T_y}{S_y}\right) + \nu_{xy}\frac{\partial}{\partial x}\left(\frac{\partial w}{\partial x} - \frac{T_x}{S_x}\right)\right] \\
M_{xy} &= -\frac{D_{33}}{2}\left[\frac{\partial}{\partial x}\left(\frac{\partial w}{\partial y} - \frac{T_y}{S_y}\right) + \frac{\partial}{\partial y}\left(\frac{\partial w}{\partial x} - \frac{T_x}{S_x}\right)\right]
\end{aligned} \tag{2.3.14}$$

By introduction of these moment-curvature relations in the differential equation above (2.3.12), this equation is not transformed into one equation in one variable w, but rather three equations (including the original differential equation (2.3.12) above) in the three variables w, T_x and T_y. These three coupled equations can be separated into three uncoupled equations, each in one of the three variables, which can then be written:

$$\begin{aligned}
[D]w &= -[M]q \\
[D]T_x &= -[N]q \\
[D]T_y &= -[P]q
\end{aligned} \tag{2.3.15}$$

where the bracketed terms represents sixth-order differential operators which can be found in Zenkert [3]. It is quite clear that solving these equations will require some effort. This can be overcome by introducing the concept of partial deflections.

2.4 Partial deflections

The plate described by the relatively complex set of differential equations in (2.3.15) can be solved by another method called partial deflections. In this method one mode of deformation is assumed at a time and the results superimposed. The deflections due to bending alone, w_b, and shear alone, w_s, are computed separately and the partial deflections are summed to get the total deflections $w = w_b + w_s$, as shown in Fig. 2.3. For a general sandwich plate, which can have face sheets made up of laminated fiber composites, this approach might be an approximation. According to Zenkert [3] the concept will be exact for isotropic sandwich plates,

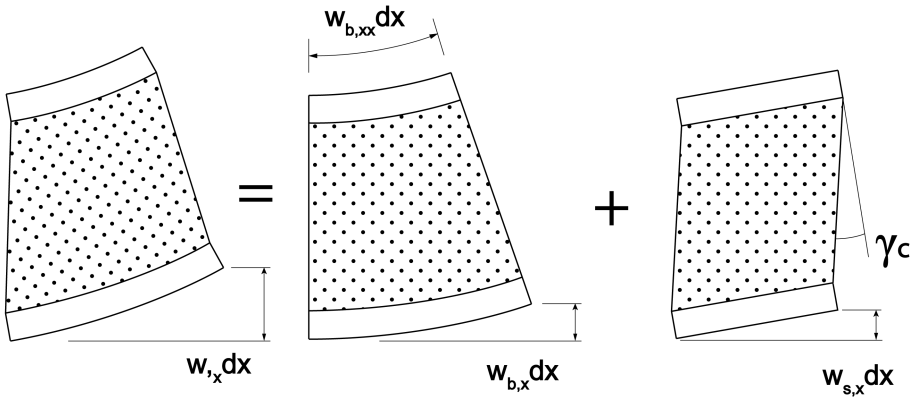


Figure 2.3: In the concept of partial deflections, total deflection (a) is the sum of bending (b) and shear (c) deflection components. The components are not coupled and can thus be computed separately.

the type of sandwich plate considered in this thesis. In short the method can be described as:

$$w = w_b + w_s \tag{2.4.1}$$

In partial deflections, bending only induces cross sectional rotations, and shearing only a sliding of the cross sections relative to each other. The consequence of this is that the moment curvature-relations can be written on the form known from Kirchhoff plate theory:

$$\begin{aligned} M_x &= -D_{11}\left[\frac{\partial^2 w_b}{\partial x^2} + \nu\frac{\partial^2 w_b}{\partial y^2}\right] \\ M_y &= -D_{22}\left[\frac{\partial^2 w_b}{\partial y^2} + \nu\frac{\partial^2 w_b}{\partial x^2}\right] \\ M_{xy} &= -D_{33}\frac{\partial^2 w_b}{\partial x\,\partial y} \end{aligned} \tag{2.4.2}$$

and the relationship between the transverse forces and shear deformations is introduced as:

$$\begin{aligned} \frac{\partial T_x}{\partial x} &= S_x\frac{\partial^2 w_s}{\partial x^2} \\ \frac{\partial T_y}{\partial y} &= S_y\frac{\partial^2 w_s}{\partial y^2} \end{aligned} \tag{2.4.3}$$

Substituting the moment-curvature relations into the differential equations (2.3.12) and (2.3.13) and combining these, it is clear that now the result is indeed the

normal fourth order differential equation in only one variable, w_b, which describes the bending only deformations:

$$D_{11}\frac{\partial^4 w_b}{\partial x^4} + 2(D_{12} + D_{33})\frac{\partial^4 w_b}{\partial x^2\,\partial y^2} + D_{22}\frac{\partial^4 w_b}{\partial y^4} = q + N_x\frac{\partial^2 w_b}{\partial x^2} + 2N_{xy}\frac{\partial^2 w_b}{\partial x\,\partial y} + N_y\frac{\partial^2 w_b}{\partial y^2} \tag{2.4.4}$$

In the same manner, the shear-sliding relations can be substituted into the same equations to get a new differential equation in only one variable, w_s, describing the shear only deformations:

$$S_x\frac{\partial^2 w_s}{\partial x^2} + S_y\frac{\partial^2 w_s}{\partial y^2} = -\left(q + N_x\frac{\partial^2 w_s}{\partial x^2} + 2N_{xy}\frac{\partial^2 w_s}{\partial x\,\partial y} + N_y\frac{\partial^2 w_s}{\partial y^2}\right) \tag{2.4.5}$$

It is worth noting that in the bending mode, the in-plane forces are functions of derivatives of w_b, while in shear mode they are functions of derivatives of w_s. This makes it possible to solve for one mode at a time and superimpose the results.

From Eqs. (2.4.4) and (2.4.5) one can see that the right hand side is similar in both equations, and from that we get the relationship between bending and shear deformations:

$$\Delta w_s = -\frac{D}{S}\Delta^2 w_b \tag{2.4.6}$$

where Δ is the Laplace operator, D is the flexural rigidity and S the shear stiffness of the plate.

As shown in Zenkert [3], for isotropic sandwich plates the concept of partial deflections can be replaced by an total deflection, w:

$$D\left[\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2\,\partial y^2} + \frac{\partial^4 w}{\partial y^4}\right] = \left[1 - \frac{D}{S}\Delta\right]\left[q + N_x\frac{\partial^2 w}{\partial x^2} + 2N_{xy}\frac{\partial^2 w}{\partial x\,\partial y} + N_y\frac{\partial^2 w}{\partial y^2}\right] \tag{2.4.7}$$

2.5 Constitutive laws

In the general orthotropic sandwich plate, the Young's modulus E, shear modulus G, Poisson's ratio ν and hence the flexural rigidity D, torsional stiffness D_{xy} and shear stiffness S may vary in the different directions. In the beginning weeks of the work in this thesis there was some uncertainty about how to arrive at accurate expressions for the stiffness, so two different methods are presented here; one from laminate theory and one found by extending the plate flexural rigidity formula for isotropic plates. They were found to yield results within 1% of each other.

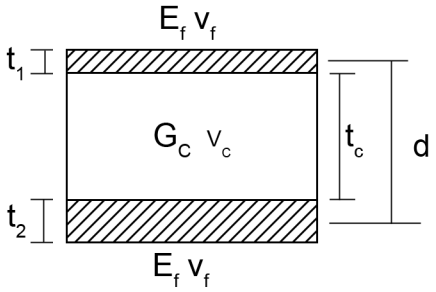


Figure 2.4: Cross-section of sandwich plate showing layer material constants and distances.

Laminate stiffness coefficients

For a laminated plate the extensional stiffness, coupling and bending stiffness matrices are:

$$
\begin{aligned}
[A] &= \sum_{k=1}^{N} [Q]^k (z_k - z_{k-1}) \\
[B] &= \frac{1}{2} \sum_{k=1}^{N} [Q]^k (z_k^2 - z_{k-1}^2) \\
[D] &= \frac{1}{3} \sum_{k=1}^{N} [Q]^k (z_k^3 - z_{k-1}^3)
\end{aligned}
\tag{2.5.1}
$$

where the [Q]-matrix is known as the reduced stiffness matrix for each layer. This must be computed for each layer from $k = 1$ up to $k = N$. The [Q]-matrix is:

$$
\mathbf{Q} = \begin{bmatrix} Q_{11} & Q_{12} & 0 \\ Q_{21} & Q_{22} & 0 \\ 0 & 0 & Q_{66} \end{bmatrix}
$$

$$
Q_{11} = \frac{E_1}{1 - \nu_{12}\nu_{21}} \qquad Q_{22} = \frac{E_2}{1 - \nu_{12}\nu_{21}} \qquad Q_{12} = \frac{\nu_{21} E_1}{1 - \nu_{12}\nu_{21}} \qquad Q_{66} = G_{12}
\tag{2.5.2}
$$

In computing the reduced stiffness matrix for a layer, one must use the properties of that layer. In the isotropic sandwich plate this means that a total of four layers will be computed, one for each face and two halves of the core. The final results of the [D]-matrix is:

$$
\mathbf{D} = \begin{bmatrix} D_{11} & D_{12} & 0 \\ D_{21} & D_{22} & 0 \\ 0 & 0 & D_{66} \end{bmatrix}
$$

$$D_{11} = \frac{1}{12}\left[\frac{E_{f1}}{1-\nu_{f1}^2}\left(3t_c^2 t_{f1} + 6t_{f1}^2 t_c + 4t_{f1}^3\right) + \frac{E_{f2}}{1-\nu_1^2}\left(3t_c^2 t_{f2} + 6t_{f2}^2 t_c + 4t_{f2}^3\right) + \frac{E_c}{1-\nu_c^2}t_c^3\right]$$

$$D_{22} = D_{11}$$

$$D_{12} = \frac{1}{12}\left[\frac{\nu_{f1}E_{f1}}{1-\nu_{f1}^2}\left(3t_c^2 t_{f1} + 6t_{f1}^2 t_c + 4t_{f1}^3\right) + \frac{\nu_{f2}E_{f2}}{1-\nu_1^2}\left(3t_c^2 t_{f2} + 6t_{f2}^2 t_c + 4t_{f2}^3\right) + \frac{\nu_c E_c}{1-\nu_c^2}t_c^3\right]$$

$$D_{21} = D_{12}$$

$$D_{66} = \frac{1}{12}\left[G_{f1}\left(3t_c^2 t_{f1} + 6t_{f1}^2 t_c + 4t_{f1}^3\right) + G_{f2}\left(3t_c^2 t_{f2} + 6t_{f2}^2 t_c + 4t_{f2}^3\right) + G_c t_c^3\right]$$

Stiffness coefficients derived from flexural rigidity

An alternative approach to the laminate stiffness is computing the flexural rigidity of a unit width of a beam and correcting this with a Poisson's ratio to include the effect of secondary curvatures. The flexural rigidity of the beam is:

$$D = \int_z E z^2 dz \tag{2.5.3}$$

where the correct E must be used in the face and core. Assuming similar thickness in the faces and defining the distance $d = t_f + t_c$, the plate stiffness can be written as:

$$D = \frac{E_f t_f^3}{6(1-\nu_f^2)} + \frac{E_f t_f d^2}{2(1-\nu_f^2)} + \frac{E_c t_c^3}{12(1-\nu_c^2)} = 2D_f + D_0 + D_c \tag{2.5.4}$$

The stiffness of an isotropic sandwich plate is the same in all in-plane directions, so the single D defined above correctly describes the plate, with $D = D_{11} = D_{22} = (D_{12} + D_{33})$. The term D_f describes the stiffness of the two face plates bending about their own axes, D_0 is the stiffness from the two face plates bending about the neutral axis of the assembled plate and D_c is the contribution to the stiffness from the core bending about its own neutral axis (which, incidentally is the same as the global neutral axis when considering similar face sheet thickness). It is seen that for isotropic sandwich plates the two ways of computing the stiffness yield approximately equal results (the difference is less than 1%), and for that reason the simpler equation in (2.5.4) has been used in the expressions for deflection and buckling. The laminate stiffness has the advantage, however, of correctly computing the stiffness of plates where the two faces are not similar; the simpler equation from Zenkert is for similar faces.

Shear stiffness

Like the flexural rigidity of the plate can be described by a single constant, D, due to the isotropic material behaviour, the shear stiffness, S, is also defined by a

single constant:

$$S = \frac{G_c\, d^2}{t_c} \tag{2.5.5}$$

where G_c is the shear modulus of the core material and t_c is the thickness of the core.

2.6 Bending of simply supported sandwich plate

The boundary conditions for a simply supported sandwich plate are:

$$w = 0,\ M_x = 0 \text{ at } x = 0 \text{ and } x = a$$
$$w = 0,\ M_y = 0 \text{ at } y = 0 \text{ and } y = b$$

and the deflection can be described by the double Fourier series which also satisfy the boundary conditions:

$$w = \sum_{n=1}^{\infty}\sum_{m=1}^{\infty} w_{mn} sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \tag{2.6.1}$$

The pressure load can also be written in the form of a double Fourier series:

$$q = \sum_{n=1}^{\infty}\sum_{m=1}^{\infty} q_{mn} sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \tag{2.6.2}$$

For a uniform pressure, the load coefficient for odd values of m and n is (for even values it is zero):

$$q_{mn} = \frac{16q}{mn\pi^2} \tag{2.6.3}$$

The deflection coefficient, w_{mn}, is found by substitution of the double series into the differential equation (2.4.7). By cancelling out equal terms, one is left with:

$$D\left[\left(\frac{m\pi}{a}\right)^4 + 2\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2 + \left(\frac{n\pi}{b}\right)^4\right] w_{mn} = \left[1 + \frac{D}{S}\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right]\right] q_{mn} \tag{2.6.4}$$

This again can be written in terms of the coefficient w_{mn}:

$$w_{mn} = \frac{q_{mn}}{D}\frac{1 + \frac{D}{S}\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right]}{\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right]} \tag{2.6.5}$$

Inserting for the coefficients w_{mn} and q_{mn}, one arrives at the expression for the transverse deflection:

$$w = \frac{16q}{\pi^6} \sum_{n=1,3,..}^{\infty} \sum_{m=1,3,..}^{\infty} \frac{\left[1 + \frac{D}{S}\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right]\right] sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}}{mnD\left[\left(\frac{m}{a}\right)^4 + 2\left(\frac{m}{a}\right)^2\left(\frac{n}{b}\right)^2 + \left(\frac{n}{b}\right)^4\right]} \tag{2.6.6}$$

Because the total deflection was assumed to be the sum of that due to bending and shear separately, the expression for deflection can conversely be split into two parts:

$$w_b = \frac{16q}{\pi^6 D} \sum_{n=1,3,..}^{\infty} \sum_{m=1,3,..}^{\infty} \frac{sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}}{mn\left[\left(\frac{mx}{a}\right)^2 + \left(\frac{ny}{b}\right)^2\right]^2} \tag{2.6.7}$$

$$w_s = \frac{16q}{\pi^4 S} \sum_{n=1,3,..}^{\infty} \sum_{m=1,3,..}^{\infty} \frac{sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}}{mn\left[\left(\frac{mx}{a}\right)^2 + \left(\frac{ny}{b}\right)^2\right]} \tag{2.6.8}$$

Effect on bending of thick face plates

Because the face sheets of the isotropic SPS-plate are made from steel the effect from the bending of the faces about their own axes had to be considered in the overall performance of the plate. In 1950, Hoff [7] included the strain energy of the faces bending about their own axes and this lead to a sixth order differential equation. Zenkert [3] has compared this theory and the normal thin-face theory. By defining a shear factor $\theta = D_0/(Sb^2)$, the lateral deflection can be written as:

$$w = \frac{16q}{\pi^6 D} \sum_{n=1,3,..}^{\infty} \sum_{m=1,3,..}^{\infty} \frac{1 + \pi^2\theta\left[\left(\frac{mb}{a}\right)^2 + n^2\right]}{mn\left[\left(\frac{mb}{a}\right)^2 + n^2\right]^2} sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \tag{2.6.9}$$

Zenkert [3] has prepared a graph which shows the effect of the thick faces compared to the thin faces in terms of the shear factor θ and the ratio D_f/D_0, shown in Fig. 2.5. In this figure, however, the shear factor is shown with the symbol ϕ. In the steel-elastomer plates considered in this thesis, the effect of using the sixth-order differential equation over the fourth-order will be negligible and the comparatively simpler fourth-order equation will yield results sufficiently accurate.

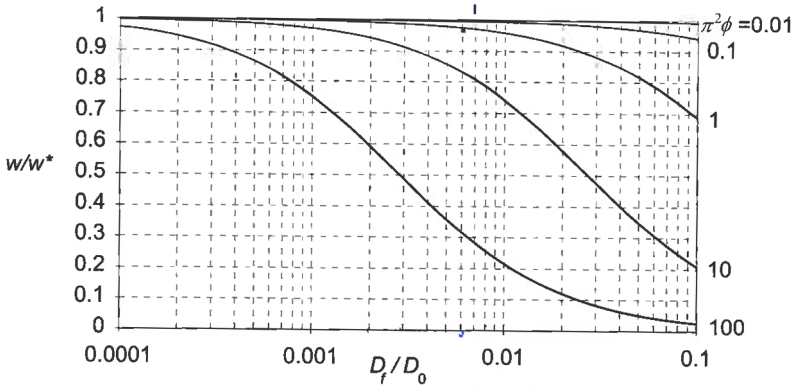


Figure 2.5: This figure shows the normalised lateral deflection of the plate when the effects of thick faces is accounted for and when it is not. It's a function of the shear factor θ and the ratio of D_f/D_0. In this figure taken from Zenkert [3], ϕ is used for the shear factor.

2.7 Buckling of simply supported sandwich plate

2.7.1 General introduction

A plate with in-plane loads can be subjected to buckling. In theory this means that for an initially perfect plate, an increasing in-plane compressive or shear load will, after some limit is reached, produce a sudden out-of-plane displacement. This limit is known as the elastic buckling load, or eigenvalue, and the corresponding deflection shape is known as buckled mode, or eigenmode. The difference in the ultimate capacity of plates compared to that of columns is that the plate will exhibit a post-critical reserve making it able to carry loads well above the elastic buckling loads. In finding the ultimate capacity of a plate, the buckling load is thus often not the governing criteria, this depends on the reduced slenderness of the plate. However, the corresponding buckling mode can be used as a least-favourable imperfection in non-linear capacity analyses. Furthermore, the elastic buckling loads are often used as a conservative limit in design of plates.

To find the buckling loads of the plate, Eq. (2.4.7) is used with $q = 0$ (no transverse pressure).

$$D\left[\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2\,\partial y^2} + \frac{\partial^4 w}{\partial y^4}\right] = \left[1 - \frac{D}{S}\Delta\right]\left[N_x\frac{\partial^2 w}{\partial x^2} + 2N_{xy}\frac{\partial^2 w}{\partial x\,\partial y} + N_y\frac{\partial^2 w}{\partial y^2}\right] \tag{2.7.1}$$

Depending on whether the plate being studied is subjected to uniaxial, biaxial or shear loading, or a combination of these, the right hand side of Eq. (2.7.1) will

differ. The same deflection shape is assumed as in the case for bending, this is substituted into the differential equation.

$$w = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} w_{mn} sin \frac{m\pi x}{a} sin \frac{n\pi y}{b} \tag{2.7.2}$$

2.7.2 In-plane compressive buckling

Introduction

Let the in-plane forces N_x and N_y equal $-P_x$ and $-P_y$, respectively, so that compressive forces act on the plate according to the assumptions made in Section 2.3 and shown in Fig. 2.2. If the plate is subjected to in-plane compression in one or two (perpendicular) directions, then the differential equation (2.7.1) will be:

$$D\left[\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2\,\partial y^2} + \frac{\partial^4 w}{\partial y^4}\right] = \left[1 - \frac{D}{S}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right]\left[-P_x\frac{\partial^2 w}{\partial x^2} - P_y\frac{\partial^2 w}{\partial y^2}\right] \tag{2.7.3}$$

Substitution of the deflection shape in Eq. (2.7.2) for w into this differential equation yields:

$$\sum_{n=1}^{\infty}\sum_{m=1}^{\infty}\left\{D\left[\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right]^2 - \left[P_x\left(\frac{m\pi}{a}\right)^2 + P_y\left(\frac{n\pi}{b}\right)^2\right]\left[1 + \frac{D}{S}\left\{\left(\frac{m\pi}{a}\right)^2 + \left(\frac{n\pi}{b}\right)^2\right\}\right]\right\} w_{mn} sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} = 0 \tag{2.7.4}$$

For this expression to be valid for all x and y, the bracketed term must equate to zero for all combinations of m and n. This equation can be used to find the elastic buckling load for uniaxial and biaxial loads, by setting the values for P_x and P_y accordingly.

Uniaxial loading

If the plate is subjected to a compressive force on two parallel sides while the two sides perpendicular to these are load free, the plate is said to be in uniaxial compression. On a plate with a non-unity aspect ratio, the lowest eigenvalue will be found from loading the two shortest edges. Any load can be represented as a product of a reference load and a scale factor, such that $P_x = \Lambda_E\, P_x^{ref}$, where the reference load can be chosen arbitrarily. By varying the scale factor the buckling

load can be found. This scale factor, Λ_E, is known as the eigenvalue. The bracketed term in the equation above will then be:

$$D\left[\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right]^2=\Lambda_E\,P_x^{ref}\left(\frac{m\pi}{a}\right)^2\left[1+\frac{D}{S}\left\{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right\}\right] \tag{2.7.5}$$

or expressed in terms of the eigenvalue:

$$\Lambda_E=\frac{D\left[\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right]^2}{P_x^{ref}\left(\frac{m\pi}{a}\right)^2\left[1+\frac{D}{S}\left\{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right\}\right]} \tag{2.7.6}$$

By re-arranging and simplifying this equation, an expression for the buckling coefficient, K, is found for the uniaxial buckling load:

$$K=\Lambda_E\frac{P_x^{ref}\,b^2}{D\,\pi^2}=\left(\frac{mb}{a}+\frac{a}{mb}\right)^2\left(1+\pi^2\theta\left[\left(\frac{mb}{a}\right)^2+1\right]\right)^{-1} \tag{2.7.7}$$

and from this equation, an expression for the lowest eigenvalue Λ_E is found:

$$\Lambda_E=\frac{D\,\pi^2}{P_x^{ref}\,b^2}\,K_{min} \tag{2.7.8}$$

The expression for the buckling factor, K, for the sandwich plate is seen to be of similar form to that of an isotropic plate, given in Brush and Almroth [6] and Băzănt [8] as:

$$K=\left(\frac{mb}{a}+\frac{a}{mb}\right)^2 \tag{2.7.9}$$

but corrected for shear-deformations with the presence of the shear factor $\theta = D/(b^2S)$ in the denominator. If the transverse shear stiffness increases, the shear factor decreases and the value of K computed with Eq. (2.7.7) approaches that computed with Eq. (2.7.9). It is also seen that the expression for the critical load in Eq. (2.7.8) is similar in form to that of the classical isotropic plate, but with the different buckling coefficient K described above. For a given plate aspect ratio, a/b, the minimum buckling load is found by varying m as shown in Fig. 2.6. The lowest eigenvalue is found by finding the smallest buckling coefficient. This is achieved by finding the combination of m and n which give the smallest K, where m and n describe the number of half-waves into which the plate buckles

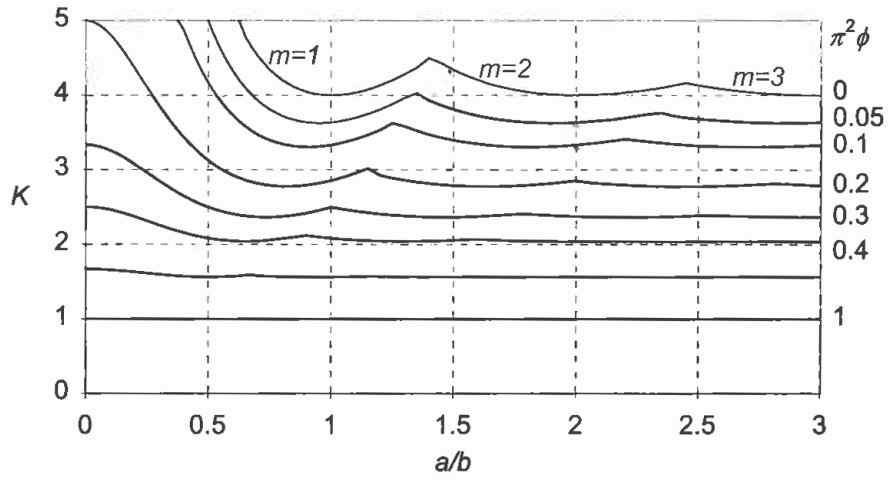


Figure 2.6: Buckling coefficient as a function of plate aspect ratio and shear factor θ. θ is printed as ϕ in this graph, from Zenkert [3]

in the longitudinal and transverse directions. From Zenkert [3] it is known that the minimum K always corresponds to a value of $n = 1$ (and indeed this has been used already in arriving at Eq. (2.7.7)), but the value of m giving the smallest K depends on the plate aspect ratio a/b. The buckling coefficient K for the isotropic sandwich plate is plotted in Fig. 2.6. The main difference in this figure and the one normally encountered for isotropic plates is that the factor K also depends on the shear factor θ and the fact that even for an aspect ratio of $a/b = 0$ the buckling load is finite. One can also see that as the shear factor increases, the different modes are less distinct from each other.

Biaxial loading

If there are loads on all four edges, the plate is said to be in biaxial loading. Let us call the two edges that were loaded in uniaxial compression analysis primary edges and the remaining two edges secondary edges. The primary edges can be either the shortest or the longest edges of the plate, depending on the problem being studied. If the load on the secondary edges is compressive, this will have a negative effect on the plate to carry loads on the primary edges. If, on the other hand, the loads on the secondary edges are tensile, the plate will be able to carry higher loads on the primary edges than in the uniaxial case. Thus, the type and magnitude of the second force is either detrimental or stabilizing on the capacity of the plate when compared to uniaxial loading. This effect can be represented in plots called interaction curves in the load space. In the case where all forces are tensional, buckling is not possible and the plate will not fail until material yield is

reached.

As for the uniaxial case, let the forces be represented by reference values and a common load scaling factor, $\Lambda_{E,bi}$. By varying this single variable for different combinations of P_x^{ref} and P_y^{ref}, the resulting eigenvalues can be used to plot different interaction curves.

$$D\left[\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right]^2 = \Lambda_{E,bi}\left[P_x^{ref}\left(\frac{m\pi}{a}\right)^2+P_y^{ref}\left(\frac{n\pi}{b}\right)^2\right]\left[1+\frac{D}{S}\left\{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right\}\right] \quad (2.7.10)$$

or expressed in terms of the eigenvalue:

$$\Lambda_{E,bi} = \frac{D\left[\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right]^2}{\left[P_x^{ref}\left(\frac{m\pi}{a}\right)^2+P_y^{ref}\left(\frac{n\pi}{b}\right)^2\right]\left[1+\frac{D}{S}\left\{\left(\frac{m\pi}{a}\right)^2+\left(\frac{n\pi}{b}\right)^2\right\}\right]} \quad (2.7.11)$$

2.7.3 In-plane shear buckling

When the plate is subjected to in-plane shear loads, N_{xy}=$-P_{xy}$, the buckling equation (2.7.1) is reduced to:

$$D\left[\frac{\partial^4 w}{\partial x^4}+2\frac{\partial^4 w}{\partial x^2\,\partial y^2}+\frac{\partial^4 w}{\partial y^4}\right] = \left[1-\frac{D}{S}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)\right]\left[-2\Lambda P_{xy}^{ref}\frac{\partial^2 w}{\partial x \partial y}\right] \quad (2.7.12)$$

There is no exact closed form solution to this equation as for the case of in-plane normal forces, however, and one has to resort to using either approximate methods or empirical formulae. An approach using an approximate method (Rayleigh-Ritz) is presented in Chapter 4, and an approach using empirical formulae found by Kuenzi and Ericksen [9] and referenced by Zenkert [3] is presented here.

Assuming that a is longer than b the general formula for a in-plane shear buckling load is:

$$P_{xy} = K\frac{\pi^2 D}{b^2} \quad (2.7.13)$$

For a simply supported plate, Kunezi and Ericksen [9] suggest the buckling coefficient, K, be computed as:

$$K = \frac{K_0}{1+\pi^2\theta\left(K_0-1-\frac{b^2}{a^2}\right)} \qquad \text{if} \qquad 0 \le \pi^2\theta \le 1+\frac{b^2}{a^2} \quad (2.7.14)$$

where

$$K_0 = \frac{16}{3} + 4\frac{b^2}{a^2} \tag{2.7.15}$$

further details and formulae for clamped edges can be found in Zenkert [3] and the development of the formulae can be found in Kuenzi and Ericksen [9].

2.7.4 Combined axial and shear buckling

When a combination of in-plane shear load and axial loads act on the plate at the same time, the following parabolic formula can be used to find the elastic buckling factor:

$$\Lambda_{E,bi+\tau} = \frac{1}{2}\left(\Lambda_{E,\tau}\right)^2 \left[-\frac{1}{\Lambda_{E,bi}} + \sqrt{\left(\frac{1}{\Lambda_{E,bi}}\right)^2 + 4\left(\frac{1}{\Lambda_{E,\tau}}\right)^2}\right] \tag{2.7.16}$$

where the eigenvalues found separately from uni- or biaxial loading $\Lambda_{E,bi}$ and shear loading $\Lambda_{E,\tau}$ are used. The parabolic formula is used in DNV CN.30.11 [1] and its background is found in Smith [10].

2.8 Elasto-plastic buckling of sandwich plate

The elastic eigenvalues found via the procedures above assume that the plate is perfectly flat and that the material is elastic. Consequently the elastic eigenvalue does not take material yield or initial imperfections into consideration. This means that the buckling loads found from elastic eigenvalue analyses can be too high for practical use, because in real plates, where there is always some amount of initial imperfection, second-order effects will lead to bending stresses in the plate and thus eventually yielding of the material. This is true for plates where the length to thickness ratio is low, so-called stocky plates. In DNV's class note, CN.30.11 Steel Sandwich Panel Construction [1], the following method is used to account for plasticity. This method takes the buckling loads found from eigenvalue analysis and corrects these loads for plastic yield in the face plate material.

First the load factor, Λ_F, which indicates the load level that will induce yielding in the face plate material is defined as:

$$\Lambda_F = \frac{\sigma_F}{\sqrt{\sigma_{10}^2 + \sigma_{20}^2 - \sigma_{10}\sigma_{20} + 3\tau_0^2}} \tag{2.8.1}$$

where σ_F is the yield stress of the face material. The applied loads N_x, N_y and N_{xy} induce the stresses σ_{10}, σ_{20} and τ_0 in the face material. These stresses are found from the formulae:

$$\sigma_{10} = \frac{N_x}{t_1 + t_2 + t_c(E_c/E_f)} \tag{2.8.2}$$

$$\sigma_{20} = \frac{N_y}{t_1 + t_2 + t_c(E_c/E_f)} \tag{2.8.3}$$

$$\tau_3 = \frac{N_{xy}}{t_1 + t_2 + t_c(E_c/E_f)} \tag{2.8.4}$$

From this load factor and the load factor found from elastic buckling analysis, Λ_E, the reduced slenderness of the plate is found:

$$\bar{\lambda} = \sqrt{\frac{\Lambda_F}{\Lambda_E}} \tag{2.8.5}$$

and from this the elasto-plastic buckling factor is found:

$$\Lambda_B = \frac{\Lambda_F}{\sqrt{1 + \lambda^4}} \tag{2.8.6}$$

This formula ensures that for any combination of loading in the load space, the elasto-plastic buckling loads will not exceed either the von Mises ellipsis or the elastic eigenvalue. The elasto-plastic buckling loads are then found by multiplying the reference loads with Λ_B.

Chapter 3

Approximate methods

3.1 Introduction

In this chapter an overview of the principle of total potential energy is presented along with two approximate theories which both spring out of this principle, the Rayleigh-Ritz and the Finite Element methods. The theory in this chapter is based on Brubak [11], Cook et. al. [12], Bažant [8] and Bergan & Syvertsen [13].

3.2 Potential energy

3.2.1 Expressions of potential energy

The potential energy of a system describes the system's ability to do work, and is made up of the internal/elastic strain energy and the potential of applied loads. That the applied loads have potential means that the loads have the capacity to do work on a system if they are displaced through a distance as the system deforms [12]. In an elastic medium, the work done by the loads is stored in the medium, and is recovered as the load is removed. The potential energy is a functional, which is a map that transforms vector functions to scalar values. Thus, according to Bergan & Syvertsen [13], a functional is a "function of functions", accepting functions as input and returning a scalar value.

In a static problem, the potential energy can be written:

$$\Pi = U + H \tag{3.2.1}$$

where U is the internal strain energy and H the load potential. If the problem is dynamic, then kinetic energy must also be included.

The *strain energy density* is the energy that must be supplied to deform a unit volume of material [12]:

$$U_0 = \int_\varepsilon \boldsymbol{\sigma}^T d\boldsymbol{\varepsilon} = \frac{1}{2}\boldsymbol{\sigma}^T\boldsymbol{\varepsilon} = \frac{1}{2}\boldsymbol{\varepsilon}^T\boldsymbol{E}\boldsymbol{\varepsilon} \tag{3.2.2}$$

and the *strain energy* of the body is found by integrating the strain energy density over the volume of the body, V:

$$U = \int_V U_0 dV = \frac{1}{2}\int_V \boldsymbol{\varepsilon}^T\boldsymbol{E}\boldsymbol{\varepsilon} dV \tag{3.2.3}$$

If the material is elastic and all six strains in the strain tensor are included, the stress $\boldsymbol{\sigma}$ and strain $\boldsymbol{\varepsilon}$ vectors each have six components and the material matrix $\mathbf{E}$ is six by six. In the Kirchhoff thin plate theory, the transverse strains are neglected, and the stress and strain vectors are of three components and the material matrix three by three. Sandwich plates are of a different theory, the Mindlin-Reissner theory. In this theory the transverse *normal* strain, but not the transverse *shear* strains, is neglected, thus giving five components in the stress and strain vectors and a five by five material matrix.
When looking at plates, one may also wish to divide up the total strain energy into bending strain energy, membrane strain energy, strain energy from stiffeners and if the plate is constrained to elastic springs, the strain energy from these springs as well.

The load potential, H, can consist of body loads and surface loads. The body loads are generally loads that affect the whole of the body, such as gravitational (accelerating) or magnetic forces, whereas the surface loads are loads that act on the surface. In its most general form the load potential H is:

$$H = -\int_V \boldsymbol{u}^T\boldsymbol{F}dV - \int_S \boldsymbol{u}^T\boldsymbol{\Phi}dS \tag{3.2.4}$$

where the volume force is obtained by integrating the volume force of an infinitesimal element over the volume, V, of the body, and the surface load is obtained by integrating over the surface, S. Added together the strain energy and load potential gives the expression for the potential energy:

$$\Pi = \frac{1}{2}\int_V \boldsymbol{\varepsilon}^T\boldsymbol{E}\boldsymbol{\varepsilon}dV - \int_V \boldsymbol{u}^T\boldsymbol{F}dV - \int_S \boldsymbol{u}^T\boldsymbol{\Phi}dS \tag{3.2.5}$$

The equation (3.2.5) is called weak form. Using this equation without first discretizing the continuous variables with a finite number of degrees of freedom, the

partial differential equation and the non-essential boundary conditions can be derived via the principle of stationary potential energy, as shown in Cook et.al. [12] and Bergan & Syvertsen [13]. Non-essential boundary conditions are the boundary conditions which include the derivatives of the continuous variables, such as stresses and moments. If there are more than one continuous variable involved, such as displacement in two directions or rotations, there will be a corresponding number of differential equations. These equations are known as Euler equations, and together with the non-essential boundary conditions they completely describe the problem. This is known as the strong form.
If, however, the continuous variables are discretized, then the potential energy will yield a set of algebraic equations. In this form the equilibrium and non-essential boundary conditions are not satisfied at all points, but only in an average sense [12].

3.2.2 Principle of virtual work

A fundamental principle in structural mechanics is the principle of virtual work or virtual displacements. This states that a system is in static equilibrium if the sum of work done by the internal and external forces, in acting through a small virtual displacement, is zero. The work done by the external forces is the negative of the potential, and the work done by the internal forces is equal to the strain energy. This can be expressed as:

$$U = W_{int} \qquad \text{and} \qquad H = -W_{ext} \tag{3.2.6}$$

and inserted into the potential energy equation (4.2.1), this equation becomes:

$$\Pi = W_{int} - W_{ext} \tag{3.2.7}$$

The principle of virtual work can be expressed as:

$$\begin{aligned} &\delta W_{int} = \delta W_{ext} \qquad \text{or} \\ &\delta\Pi = \delta U - \delta W \quad = \quad \int_V \boldsymbol{\varepsilon} \boldsymbol{E} \delta\boldsymbol{\varepsilon} dV - \int_V \delta \boldsymbol{u}^T \boldsymbol{F} dV - \int_S \delta \boldsymbol{u}^T \boldsymbol{\Phi} dS = 0 \end{aligned} \tag{3.2.8}$$

This leads to an important fact, that for a conservative[1] system the internal strain energy and external work are equal.

[1] That the system is conservative means that the work done in deforming the system is only dependent on the initial and deformed configurations of the system, not the path taken between the two configurations

3.2.3 Principle of stationary potential energy

From the principle of virtual work another important principle in structural mechanics can be derived, the principle of stationary potential energy (PSPE). It states that *all configurations which make the potential energy stationary with respect to small admissible variations of displacement will satisfy the equations of equilibrium* [12]. If this stationary point of Π is also a minimum, the configuration is a stable one.

To illustrate, consider a spring with stiffness k, fixed at one end and with a load P on the other end that works through a distance D. The potential energy of the system is:

$$\Pi = U + H = \frac{1}{2}kD^2 - PD \tag{3.2.9}$$

By finding a stationary value of Π, an expression for the equilibrium state can be found:

$$\frac{\partial \Pi}{\partial D} = 0 \qquad \rightarrow \qquad \partial \Pi = (kD - P)dD = 0 \tag{3.2.10}$$

Because the equation must hold for all small admissible variations of displacement dD, the expression in the brackets must equal zero. From this we get the equilibrium equation:

$$D_{eq} = \frac{P}{k} \tag{3.2.11}$$

This expression says nothing about the stability of the equilibrium. It could very well be an unstable configuration. However, by finding the second variation of the potential energy, we see that it is equal to the spring stiffness k, thus positive and the configuration is stable. For a system with linear-elastic material and conservative loads this is identical to using the Principle of Minimum Potential Energy (PMPE). Although a principle that is much less applicable than the PSPE, it is nonetheless the same for this kind of system.

Using the principle on the potential energy equation (3.2.5) we get:

$$\delta\Pi = \frac{1}{2}\int_V \left(\delta\boldsymbol{\varepsilon}^T \boldsymbol{E}\boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T \boldsymbol{E}\delta\boldsymbol{\varepsilon}\right) dV - \int_V \delta\boldsymbol{u}^T \boldsymbol{F} dV - \int_S \delta\boldsymbol{u}^T \boldsymbol{\Phi} dS = 0 \tag{3.2.12}$$

which will lead to the Euler equations and the non-essential boundary conditions.

3.3 Rayleigh-Ritz method

The Rayleigh-Ritz method gets its name from Lord Rayleigh and Walter Ritz. In the 1870s Lord Rayleigh used one degree of freedom to approximate the natural vibration frequencies in vibration studies, while Ritz expanded the method when he used a series of approximating functions (multiple d.o.f.) to study eigenvalue problems in 1909 [12].

The above example of the bar is very good to show the power of the potential energy. It is also a very simple example. Only the deformation at the end of the spring is needed to describe the state of that system, the same cannot be said of the deformation of single point when looking, for instance, on a plate. In the more general systems, we need to know the deformations at many points in order to describe the state of the deformation. The Euler equations that can be derived from the potential energy describe the continuous variable (deformation) at *all* points in the body, but finding solutions to these equations can be challenging except for relatively simple problems.

The problem of finding a solution to the differential equations can be overcome by approximating the continuous variable as a linear combination of functions. Each function is assigned an amplitude, and the solution is now to find the amplitudes of each function such that the potential energy reaches a minimum. The amplitudes of these functions are called degrees of freedom (d.o.f.). All the degrees of freedom are collected in a displacement vector $\mathbf{a}$. The use of the functions makes the potential energy a functional of expressions containing these amplitudes, rather than the single continuous variable:

$$\Pi = \Pi(a_1, a_2, a_3, ..., a_m) \tag{3.3.1}$$

and the requirement of equilibrium is now that the potential energy be stationary with respect to all the degrees of freedom:

$$\frac{\partial \Pi}{\partial a_1} = 0, \frac{\partial \Pi}{\partial a_2} = 0, \frac{\partial \Pi}{\partial a_3} = 0, ..., \frac{\partial \Pi}{\partial a_m} = 0 \qquad \text{or} \qquad \frac{\partial \Pi}{\partial \mathbf{a}} = \mathbf{0} \tag{3.3.2}$$

The functions used in the approximation are called trial or shape functions, and are Fourier series which, with sufficient number of terms in the series, can approximate the deflection and rotations. An example of a trial function is:

$$w = \sum_{i=1}^{M} a_i f(x)_i \tag{3.3.3}$$

where $f(x)_i$ is the approximating functions and a_i is the amplitude of the function. Thus, the problem has been reduced from finding a solution to the differential equation, to solving a set of M algebraic equations in the unknown amplitudes a_i.

This also brings on a major consequence of the Rayleigh-Ritz method. Because the exact and smooth deflection shape is approximated by a combination of displacement shapes, the system is made stiffer than it really is. This happens because the shape functions constrain the system to deform in a manner similar, but not, exactly as it wants. If the potential energy functional is quadratic, as is the case for linear elastic systems, convergence will always be from above [8]. As a result, deflections are under-predicted with the Rayleigh-Ritz method, while buckling loads are over-predicted. In the general case the relative difference between the approximated and exact values is decreasing as more terms are used in the trial functions. An important exception, however, is when the exact deflection shape appears as a term in the series. This is the case with uniaxial and biaxial loading of a simply supported plate, but not in shear loading. The reason is that the deflection shape of the plate due to uniaxial or biaxial loading can be described exactly by trigonometric functions. In this case the deflections and buckling loads will be exact. For shear loading, however, there is no single function that can describe the deflection shape. Here the method will use the shape functions to create an approximation of the true deformations, selecting the amplitudes of each of these functions so that the potential energy reaches a stationary point. The same is the case with clamped plates, or plates with elastic springs at the boundaries.

The trial functions cannot be chosen arbitrarily. In Bažant [8] the theorem of Ritz is given:

> The limit for the approximation $w_M(x) = \sum_{i=1}^{M} a_i f(x)_i \quad \text{for} \quad M \to \infty$ is the exact solution $w(x)$ if the system of chosen functions satisfies the conditions:
>
> 1. Functions $f(x)_i$ are linearly independent
>
> 2. Functions $f(x)_i$ form a complete system of functions
>
> 3. Functions $f(x)_i$ satisfy the essential boundary conditions

They must be admissible, complete and they must converge as the series are expanded. That they are compatible means that they must satisfy internal compatibility (continuous and smooth shapes) and they must satisfy the essential boundary conditions, which are displacements and rotations at the boundaries. Completeness is satisfied when the displacements and their derivatives can be matched

closely if enough terms are used in the trial functions. Convergence is of course also a requirement, as the whole purpose of the method is to approximate an exact value.

When the correct trial functions have been found they must be inserted into the potential energy expression. The potential energy is no longer an expression of the unknown continuous variables, but the generalized degrees of freedom. If the problem is described by more than one unknown, such as is the case with the sandwich plate in this thesis, there must be one trial function for each unknown. Also, if the problem is in more than one dimension, the trial functions must be functions of the variables describing these dimensions, such as the Cartesian or polar co-ordinates. While the procedure of solving the Rayleigh-Ritz model thus becomes more complicated as more unknowns and variables are used, the procedure still remains the same.

An example of potential energy using two unknowns, w_1 and w_2, approximated by the trial functions [13]:

$$w_1 = \sum_{i=1}^{M} a_i f(x)_i \qquad\qquad w_2 = \sum_{j=1}^{N} b_j f(x)_j \tag{3.3.4}$$

$$\Pi(w_1, w_2) \rightarrow \Pi(a_1, a_2, ..., a_N, b_1, b_2, ..., b_N) \tag{3.3.5}$$

The condition for buckling is that the potential energy has a stationary value:

$$\delta\Pi = \delta U + \delta H = 0 \tag{3.3.6}$$

$$\delta\Pi = \sum_{i=1}^{M} \frac{\partial \Pi}{\partial a_i}\delta a_i + \sum_{j=1}^{N} \frac{\partial \Pi}{\partial b_j}\delta b_i = 0 \tag{3.3.7}$$

Because the equilibrium must hold for all small variations of a_i and b_j, the problem is reduced to requiring all the partial derivatives equal to zero:

$$\begin{aligned}
\frac{\partial \Pi}{\partial a_1} &= \frac{\partial U}{\partial a_1} + \frac{\partial H}{\partial a_1} = 0 \\
&\vdots \quad\quad \vdots \quad\quad \vdots \quad\quad \vdots \\
\frac{\partial \Pi}{\partial a_M} &= \frac{\partial U}{\partial a_M} + \frac{\partial H}{\partial a_M} = 0 \\
\frac{\partial \Pi}{\partial b_1} &= \frac{\partial U}{\partial b_1} + \frac{\partial H}{\partial b_1} = 0 \\
&\vdots \quad\quad \vdots \quad\quad \vdots \quad\quad \vdots \\
\frac{\partial \Pi}{\partial b_N} &= \frac{\partial U}{\partial b_N} + \frac{\partial H}{\partial b_N} = 0
\end{aligned}$$

If all the generalized degrees of freedom are collected in a vector $\mathbf{A}$, the problem can be expressed on vector form:

$$\frac{\partial \Pi}{\partial \mathbf{A}} \delta \mathbf{A} = \left(\frac{\partial U}{\partial \mathbf{A}} + \frac{\partial H}{\partial \mathbf{A}} \right) \delta \mathbf{A} = \mathbf{0} \tag{3.3.8}$$

From this equation we can formulate the eigenvalue equation used in buckling analysis:

$$(\mathbf{K}^M - \Lambda \mathbf{K}^G)\mathbf{A} = \mathbf{0} \tag{3.3.9}$$

where $\mathbf{K}^M$ and $\mathbf{K}^G$ are the material and geometric stiffness matrices, and Λ and $\mathbf{A}$ is the eigenvalue and the corresponding vector of displacements (eigenvector). The task is now to solve this eigenproblem and find the eigenvalue.

3.4 Finite Element method

3.4.1 Introduction

The distribution of displacements in a structure is known as a field problem and is described by the differential equations. Just like the Rayleigh-Ritz method is a way to avoid solving the differential equation by transforming it into a set of algebraic

equations, the Finite Element method is another method with the same goal. Indeed, the Rayleigh-Ritz and Finite Element methods have much in common. Besides the Rayleigh-Ritz method from the early 20th century, other methods had been developed before the Second World War by mathematicians in order to solve differential equations, such as Finite Difference methods or weighted residual methods (i.e. Galerkin). Comprehensive lists of these developments are given in Cook et. al. [12] and Zienkiewicz & Taylor [14]. After the Second World War, aerospace engineers in the United States developed the Finite Element method from an engineering rather than a mathematical standpoint. Some years later, in the 1960s, the foundation between the Finite Element method earlier methods such as Rayleigh-Ritz was established, and the method that was previously considered an engineer's tool was given a sound mathematical foundation.

The main features of the Finite Element method is that it can be used on very complex geometries, non-linear deformations and materials and other iterative problems. The essence of the method is that in order to find a solution to the field problems, the structure is divided into a finite number of elements. The name *finite* is to distinguish them from the infinitesimal elements of calculus. Although the geometry of the structure can be very complex, the division into elements is made so that each element is of simple shape. This makes it possible to formulate equilibrium equations for each element in matrix form.

The elements are connected together through nodes, and the distribution of elements through the structure is called a mesh. The nodes of each element are numbered and then given a global number in the structure mesh. In this way, a global stiffness matrix can be built by adding in the stiffness matrices of the elements based on the placement of each element in the mesh. When the global stiffness matrix has been established, boundary conditions and loads are prescribed.

The problem is now on the form:

$$\mathbf{KD} = \mathbf{R} \tag{3.4.1}$$

where $\mathbf{K}$ is the global stiffness matrix, $\mathbf{D}$ is the vector with degrees of freedom (displacements and rotations) and $\mathbf{R}$ is the global load vector. The problem is solved for the unknown displacements and loads.

3.4.2 Method

In order to get to the global equilibrium equation in Eq. (3.4.1), one has to assemble the global stiffness matrix, the vector with global degrees of freedom

and the vector with total loads. These are assembled from the element matrices and vectors. In structural mechanics, the formulation of elements is most often displacement based. That means that it is the displacements that are approximated from a set of trial functions, much like in the Rayleigh-Ritz method, and the element stiffness matrix is developed from the approximated displacements. Displacement based elements always satisfy the compatibility equations [12]. An important distinction from the Rayleigh-Ritz method, where the field variables are approximated over the whole structural domain, is that in the Finite Element method this approximation is only defined inside each element. This leads to the fact that the variables are continuous over each element, but not necessarily across element borders [12].

The main difference in the formulation of plate elements which accounts for transverse shear deformation, called Mindlin-Reissner elements, and plate elements that disregard these deformations, called Kirchhoff elements, is that in the latter the complete state of deformation can be described by a single field, the mid-surface deflection w, while in the former the state of deformation is w and also the rotation of cross sections with respect to the mid-surface, ϕ_x and ϕ_y. That makes the matrices and vectors larger, but the process of element formulation is completely the same for the two elements.

First, the field variables (displacement and rotations) are approximated from trial functions. By interpolating each continuous variable from a set of trial functions, the field is satisfied in a number of discrete points, namely the nodes of the mesh. For a plate with shear deformation theory, the interpolation of the fields is:

$$\begin{aligned} w &= N_1 w_1 + N_2 w_2 + ... + N_i w_i \\ \phi_x &= N_1 \phi_{x1} + N_2 \phi_{x2} + ... + N_i \phi_{xi} \\ \phi_y &= N_1 \phi_{x1} + N_2 \phi_{x2} + ... + N_i \phi_{xi} \end{aligned} \tag{3.4.2}$$

where each N_i is a trial function. The trial functions must be chosen with care, so that they are able to accurately describe all deformations and rigid body translations and rotations. The field must not omit the lowest-order terms, and must be balanced (i.e. not favour one coordinate axis to another)[12]. The interpolated fields can be written on a matrix-vector form (where n is the number of nodes pr. element):

$$\begin{bmatrix} w \\ \phi_x \\ \phi_y \end{bmatrix} = \sum_{i=1}^{n} \begin{bmatrix} N_i & 0 & 0 \\ 0 & N_i & 0 \\ 0 & 0 & N_i \end{bmatrix} \begin{bmatrix} w_i \\ \phi_{xi} \\ \phi_{yi} \end{bmatrix} \qquad \Leftrightarrow \qquad \mathbf{u} = \mathbf{N}\mathbf{d}$$

The principle of virtual work can be used to formulate the element stiffness matrix and load vector. Stated in Section 3.2.2, the principle says that for any admissible

virtual displacement of a system in equilibrium, the sum of external and internal work is zero:

$$\int_V \delta\boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dV - \int_V \delta\mathbf{u}^T \mathbf{F} dV - \int_S \delta\mathbf{u}^T \boldsymbol{\Phi} dS = 0 \tag{3.4.3}$$

The strains are derived from the displacements via the known relation:

$$\boldsymbol{\varepsilon} = \partial\mathbf{u} = \partial\mathbf{N}\mathbf{d} = \mathbf{B}\mathbf{d} \tag{3.4.4}$$

where ∂ is a differential operator matrix, and $\mathbf{B}$ is known as the strain-displacement matrix. From this we get the virtual displacements:

$$\delta\boldsymbol{\varepsilon}^T = \delta\mathbf{d}^T\mathbf{B}^T \qquad \text{and} \qquad \delta\mathbf{u}^T = \delta\mathbf{d}^T\mathbf{N}^T \tag{3.4.5}$$

which, when inserted into the expression for virtual work yield the equation:

$$\delta\mathbf{d}^T \left(\int_V \mathbf{B}^T\mathbf{E}\mathbf{B} dV\,\mathbf{d} - \int_V \mathbf{N}^T\mathbf{F} dV - \int_S \mathbf{N}^T\boldsymbol{\Phi} dS \right) = 0 \tag{3.4.6}$$

where Hooke's law on matrix form $\boldsymbol{\sigma} = \mathbf{E}\boldsymbol{\varepsilon}$ has been used to insert for $\boldsymbol{\sigma}$. For this equation to be valid for any small virtual displacement $\delta\mathbf{d}$, the expression inside the brackets must equal zero, or:

$$\mathbf{k}\mathbf{d} = \mathbf{r} \tag{3.4.7}$$

where the element stiffness matrix and load vectors are:

$$\mathbf{k} = \int_V \mathbf{B}^T\mathbf{E}\mathbf{B} dV \qquad \text{and} \qquad \mathbf{r} = \int_V \mathbf{N}^T\mathbf{F} dV + \int_S \mathbf{N}^T\boldsymbol{\Phi} dS \tag{3.4.8}$$

where dV is the volume and dS the surface of the element in question.
By expanding the element stiffness matrices and vectors to structure size, and using the $\mathbf{a}$ matrix to couple local degrees of freedom to global, we get the global relation in Eq. (3.4.1):

$$\mathbf{K} = \sum_{i=1}^{N} \mathbf{a}_i^T\mathbf{k}_i\mathbf{a}_i \qquad \mathbf{D} = \sum_{i=1}^{N} \mathbf{a}_i^T\mathbf{d}_i \qquad \mathbf{R} = \sum_{i=1}^{N} \mathbf{a}_i^T\mathbf{r}_i \tag{3.4.9}$$

where N is the number of elements in the mesh.

The problem is not yet completely defined, however. If no constraints are imposed on Eq. (3.4.1), the stiffness matrix $\mathbf{K}$ will be singular and the solution will fail. By imposing sufficient number of constraints known as boundary conditions on the global degrees of freedom, the stiffness matrix can be re-arranged so that one part of it relates to known degrees of freedom, and the other part to unknown degrees of freedom. By a process of static condensation, the unknown degrees of freedom are found and the field problem is solved [12].

3.4.3 Eigenvalue problem and buckling in FEA

An idealized perfect and initially flat plate is susceptible to buckling when in-plane forces are present. In buckling, the in-plane forces will reach a limit and large transverse deformations will quickly develop. This happens because the plate has far greater membrane stiffness than bending stiffness, and large membrane strain energy can be stored with small deformations [12]. When the buckling occurs, the large membrane strain energy must be absorbed by bending strain energy, but because the bending stiffness is much smaller, the deformations conversely become much larger.

Buckling is an eigenvalue problem, in which the material stiffness matrix is augmented by a geometric stiffness matrix $\mathbf{K}^G$. The global geometric stiffness matrix is assembled from element matrices, just as the material stiffness matrix $\mathbf{K}^M$. According to Cook et. al [12], the first step is to load the structure with an arbitrary reference load $\mathbf{R}_{ref}$ which will yield a geometric stiffness matrix $\mathbf{K}^G_{ref}$. Any other load can be expressed by the reference load by multiplying it with a scalar, $\mathbf{R} = \Lambda\mathbf{R}_{ref}$, and the geometric stiffness matrix for any load level is found the same way $\mathbf{K}^G = \Lambda\mathbf{K}^G_{ref}$. At bifurcation buckling the loads will remain the same but the lateral deflection will suddenly increase. By comparing the configuration at buckling relative to the reference configuration, we get the two equations:

$$\begin{aligned} \left(\mathbf{K}^M + \Lambda_{cr}\mathbf{K}^G_{ref}\right)\mathbf{D}_{ref} &= \Lambda_{cr}\mathbf{R}_{ref} \qquad \text{and} \\ \left(\mathbf{K}^M + \Lambda_{cr}\mathbf{K}^G_{ref}\right)\left[\mathbf{D}_{ref} + \delta\mathbf{D}\right] &= \Lambda_{cr}\mathbf{R}_{ref} \end{aligned} \tag{3.4.10}$$

where subtraction of the first from the second equation yields the eigenvalue problem:

$$\left(\mathbf{K}^M + \Lambda_{cr}\mathbf{K}^G_{ref}\right)\delta\mathbf{D} = \mathbf{0} \tag{3.4.11}$$

and from this equation the critical loads can be found as:

$$\mathbf{R}_{cr} = \Lambda_{cr}\mathbf{R}_{ref} \tag{3.4.12}$$

Chapter 4

Rayleigh-Ritz model for buckling

4.1 Introduction

The Rayleigh-Ritz model derived in this chapter extends the normal thin plate theory to a model which can accurately model plates where transverse shear deformations might be of importance. By approximating not only the transverse deflection, w, but also the cross-sectional rotations, γ_{xz} and γ_{yz}, from Fourier series, these effects are accounted for. The model is primarily developed with sandwich plates in mind, but can also be used with thick isotropic steel plates. Both simply supported and clamped plates have been considered. The derived model has been implemented in the computer language FORTRAN.

4.2 Rayleigh-Ritz model

4.2.1 General model - simply supported plate

The total potential energy comes from the internal strain energy, U, and the potential of external loads, W, as described above. The total potential energy is written:

$$\Pi = U - W \tag{4.2.1}$$

For the particular case of a sandwich plate, the strain energy comes from the moments doing work through a curvature as the plate bends and transverse shear forces sliding the cross sections relative to each other in shear deformations. The work done by the moment M_x in bending a plate element $dx\ dy$ through the curvature $w_{,xx}$ is:

$$\frac{1}{2}M_x\left(-\frac{\partial^2 w}{\partial x^2}\right)dx\,dy \quad = \quad \frac{1}{2}M_x\left(\frac{M_x}{D_x}-\nu_{yx}\frac{M_y}{D_y}\right)dx\,dy \tag{4.2.2}$$

and similarly for moment M_y working through the curvature $w_{,yy}$. The twisting moment M_{xy} working through the twisting curvature $w_{,xy}$ is:

$$2 \times \frac{1}{2} M_{xy} \left(-\frac{\partial^2 w}{\partial x \partial y} \right) dx\, dy \quad = \quad \frac{M_{xy}^2}{D_{xy}} dx\, dy \tag{4.2.3}$$

In this shear deformation theory, the contributions to the strain energy from transverse shear forces, T_x and T_y, in working through the shear angles must also be considered:

$$\frac{1}{2} \frac{T_x^2}{S_x} dx\, dy \qquad \text{and} \qquad \frac{1}{2} \frac{T_y^2}{S_y} dx\, dy \tag{4.2.4}$$

In order to get the total strain energy of the plate, the preceding work equations must be added together and the resulting equation integrated over the plate:

$$\begin{aligned} U_{plate} &= \frac{1}{2} \int_{y=0}^{b} \int_{x=0}^{a} \left[M_x \left(\frac{M_x}{D_x} - \nu_{yx} \frac{M_y}{D_y} \right) + M_y \left(\frac{M_y}{D_y} - \nu_{xy} \frac{M_x}{D_x} \right) + 2 \frac{M_{xy}^2}{D_{xy}} + \frac{T_x^2}{S_x} + \frac{T_y^2}{S_y} \right] dxdy \\ &= \frac{1}{2} \int_{y=0}^{b} \int_{x=0}^{a} \left[\frac{M_x^2}{D_x} - \left(\frac{\nu_{xy}}{D_x} + \frac{\nu_{yx}}{D_y} \right) M_x\, M_y + \frac{M_y^2}{D_y +} + 2 \frac{M_{xy}^2}{D_{xy}} + \frac{T_x^2}{S_x} + \frac{T_y}{S_y} \right] dxdy \end{aligned}$$

By using the moment curvature relations in Eq. (2.3.14) and using the facts that $D_x >> \nu^2\, D_x$ and that $\gamma_{xz} = T_x / S_x$, the strain energy can be written:

$$\begin{aligned} U_{plate} = \frac{1}{2} \int_{y=0}^{b} \int_{x=0}^{a} \Bigg[& D_x \left(\frac{\partial}{\partial x} \left(\frac{\partial w}{\partial x} - \gamma_{xz} \right) \right)^2 + (D_x \nu_{yx} D_y \nu_{xy}) \frac{\partial}{\partial x} \left(\frac{\partial w}{\partial x} - \gamma_{xz} \right) \frac{\partial}{\partial y} \left(\frac{\partial w}{\partial y} - \gamma_{yz} \right) \\ & + D_y \left(\frac{\partial}{\partial y} \left(\frac{\partial w}{\partial y} - \gamma_{yz} \right) \right)^2 + \frac{D_{xy}}{2} \left(\frac{\partial}{\partial x} \left(\frac{\partial w}{\partial y} - \gamma_{yz} \right) + \frac{\partial}{\partial y} \left(\frac{\partial w}{\partial x} - \gamma_{xz} \right) \right)^2 \\ & + T_x \gamma_{xz} + T_y \gamma_{yz} \Bigg] dx\, dy \end{aligned} \tag{4.2.5}$$

The potential of external loads is:

$$W = \frac{1}{2} \int_{y=0}^{b} \int_{x=0}^{a} \left[-2qw + N_x \left(\frac{\partial w}{\partial x} \right)^2 + 2 N_{xy} \left(\frac{\partial w}{\partial x} \right) \left(\frac{\partial w}{\partial y} \right) + Ny \left(\frac{\partial w}{\partial y} \right)^2 \right] \tag{4.2.6}$$

4.2.2 Elastic springs on the boundary - clamped plate

The model above describes a simply supported plate. That is, the edges are free to rotate at the four boundaries. If one wishes to consider a clamped plate, this can

be achieved by stiffening the plate with elastic springs along each of the four edges. By using very high spring stiffness coefficients, the rotations will be restrained and the plate will behave as if it were clamped. In the Rayleigh-Ritz method, the energy stored in the springs will be computed much like the stiffness matrix of the plate itself and be added to the total potential energy:

$$\Pi = U_{plate} + U_{springs} - W \tag{4.2.7}$$

where the energy from the springs is the sum of the energy of the springs at each boundary:

$$U_{springs} = U^s_{x=0} + U^s_{x=a} + U^s_{y=0} + U^s_{y=b} \tag{4.2.8}$$

The energy associated with the spring at the boundary $x = 0$ is written:

$$U^s_{x=0} = \frac{1}{2}\int_0^b k_{x0}\left(\left[\frac{\partial w}{\partial x} - \gamma_{xz}\right]\left[\frac{\partial w}{\partial x} - \gamma_{xz}\right]\right)_{x=0} dy \tag{4.2.9}$$

and similarly for the three other edges:

$$U^s_{x=a} = \frac{1}{2}\int_0^b k_{xa}\left(\left[\frac{\partial w}{\partial x} - \gamma_{xz}\right]\left[\frac{\partial w}{\partial x} - \gamma_{xz}\right]\right)_{x=a} dy \tag{4.2.10}$$

$$U^s_{y=0} = \frac{1}{2}\int_0^a k_{y0}\left(\left[\frac{\partial w}{\partial y} - \gamma_{yz}\right]\left[\frac{\partial w}{\partial y} - \gamma_{yz}\right]\right)_{y=0} dx \tag{4.2.11}$$

$$U^s_{y=b} = \frac{1}{2}\int_0^a k_{yb}\left(\left[\frac{\partial w}{\partial y} - \gamma_{yz}\right]\left[\frac{\partial w}{\partial y} - \gamma_{yz}\right]\right)_{y=b} dx \tag{4.2.12}$$

where the spring constant k_{xa} is the stiffness of the spring located at the boundary where $x = a$, and similarly for the springs located on the other three edges.

4.2.3 Displacement functions

The problem described by the Eqs. (4.2.1), (4.2.5), (4.2.6) and (4.2.8) can be approximated by the Rayleigh-Ritz method. By assuming displacement functions, the solution will converge to the exact solution as more degrees of freedom are added. In this thesis, the following displacement functions are used:

$$\begin{aligned} w &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\, sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\ \gamma_{xz} &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} B_{mn}\, cos\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\ \gamma_{yz} &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} C_{mn}\, sin\frac{m\pi x}{a} cos\frac{n\pi y}{b} \end{aligned} \tag{4.2.13}$$

These displacement functions must be differentiated to be inserted into the expression for potential energy. These differentiated expressions are found in Appendix A.1.

The Rayleigh-Ritz method rarely yields the exact solution. Rather, the computed eigenvalues converge from above toward the exact eigenvalues as more terms in the displacement series are added. These functions are approximated by adding a finite number of terms in the series, from 1 up to M, N, P and Q, respectively.

4.2.4 Potential energy for simply supported plate

The displacement functions and their derivatives are inserted into the expressions for strain energy and load potential. This procedure is shown in Appendix A.2, and using the integrals in Appendix A.3 the expression for the potential energy of the plate is:

$$\begin{aligned}
U = \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\Bigg[& D_x\left[A_{mn}^2\left(\frac{m\pi}{a}\right)^4 - 2A_{mn}B_{mn}\left(\frac{m\pi}{a}\right)^3 + B_{mn}^2\left(\frac{m\pi}{a}\right)^2\right] \\
& + (D_x\nu_{yx} + D_y\nu_{xy})\left[A_{mn}^2\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2 - A_{mn}C_{mn}\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)\right. \\
& \left. - A_{mn}B_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)^2 + B_{mn}C_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\right] \\
& + D_y\left[A_{mn}^2\left(\frac{n\pi}{b}\right)^4 - 2A_{mn}C_{mn}\left(\frac{n\pi}{b}\right)^3 + C_{mn}^2\left(\frac{n\pi}{b}\right)\right] \\
& + \frac{D_{xy}}{2}\left[A_{mn}^2\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2 - 2A_{mn}C_{mn}\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right) + C_{mn}^2\left(\frac{m\pi}{a}\right)^2\right. \\
& + 2\left(A_{mn}^2\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2 - A_{mn}B_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)^2\right. \\
& \left.\left. - A_{mn}C_{mn}\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right) + B_{mn}C_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\right) \right. \\
& \left. + A_{mn}^2\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2 - 2A_{mn}B_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)^2 + B_{mn}^2\left(\frac{n\pi}{b}\right)^2\right] \\
& + S_xB_{mn}^2 + S_yC_{mn}^2\Bigg]\frac{ab}{4}
\end{aligned} \tag{4.2.14}$$

$$W = \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\left[N_x\, A_{mn}^2\left(\frac{m\pi}{a}\right)^2 + N_y\, A_{mn}^2\left(\frac{n\pi}{b}\right)^2\right]\frac{ab}{4} \tag{4.2.15}$$
$$+\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q} A_{mn}A_{pq}N_{xy}\left(\frac{m\pi}{a}\right)\left(\frac{q\pi}{b}\right)\left[\frac{a(1-(-1)^m(-1)^p)p}{\pi(p^2-m^2)}\right]\left[\frac{b(1-(-1)^n(-1)^q)n}{\pi(n^2-q^2)}\right] I_{mp}I_{nq}$$

where I_{mp} is a matrix which ensures that the only non-zero terms are the ones where m is *not* equal to p, and similarly for I_{nq}.

4.2.5 Potential energy for clamped plate

The elastic springs on the boundary will stiffen the plate. When inserting for w, γ_{xz} and γ_{yz}, the expressions in Eqs. (4.2.9) to (4.2.12) are:

$$U_{springs} = \frac{k_{x0}}{2}\int_0^b \left(\sum_{m=1}^{M}\sum_{n=1}^{N}\left[A_{mn}\left(\frac{m\pi}{a}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b} - B_{mn}cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right]\right.$$
$$\left.\sum_{p=1}^{P}\sum_{q=1}^{Q}\left[A_{pq}\left(\frac{p\pi}{a}\right)cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} - B_{pq}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b}\right]\right)_{x=0} dy$$
$$+\frac{k_{xa}}{2}\int_0^b \left(\sum_{m=1}^{M}\sum_{n=1}^{N}\left[A_{mn}\left(\frac{m\pi}{a}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b} - B_{mn}cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right]\right.$$
$$\left.\left[\sum_{p=1}^{P}\sum_{q=1}^{Q}A_{pq}\left(\frac{p\pi}{a}\right)cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} - B_{pq}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b}\right]\right)_{x=a} dy$$
$$+\frac{k_{y0}}{2}\int_0^a \left(\sum_{m=1}^{M}\sum_{n=1}^{N}\left[A_{mn}\left(\frac{n\pi}{b}\right)cos\frac{m\pi x}{a}sin\frac{l\pi y}{b} - C_{mn}cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right]\right.$$
$$\left.\left[\sum_{p=1}^{P}\sum_{q=1}^{Q}A_{pq}\left(\frac{q\pi}{b}\right)cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} - C_{pq}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b}\right]\right)_{y=0} dx$$
$$+\frac{k_{yb}}{2}\int_0^a \left(\sum_{m=1}^{M}\sum_{n=1}^{N}\left[A_{mn}\left(\frac{n\pi}{b}\right)cos\frac{m\pi x}{a}sin\frac{l\pi y}{b} - C_{mn}cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right]\right.$$
$$\left.\left[\sum_{p=1}^{P}\sum_{q=1}^{Q}A_{pq}\left(\frac{q\pi}{b}\right)cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} - C_{pq}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b}\right]\right)_{y=b} dx$$

Evaluating the integrals at their limits and using the condition in Eq. (A.3.1) one gets:

$$\begin{aligned}
U_{springs} = &\frac{bk_{x0}}{4}\sum_{m=1}^{M}\sum_{m=1}^{N}\sum_{p=1}^{P}\left(A_{mn}\left(\frac{m\pi}{a}\right)-B_{mn}\right)\left(A_{pn}\left(\frac{p\pi}{a}\right)-B_{pn}\right)\\
&+\frac{bk_{xa}}{4}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{p=1}^{P}\left(A_{mn}\left(\frac{m\pi}{a}\right)-B_{mn}\right)\left(A_{pn}\left(\frac{p\pi}{a}\right)-B_{pn}\right)cos(m\pi)cos(p\pi)\\
&+\frac{ak_{y0}}{4}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{q=1}^{Q}\left(A_{mn}\left(\frac{n\pi}{b}\right)-C_{mn}\right)\left(A_{mq}\left(\frac{q\pi}{b}\right)-C_{mq}\right)\\
&+\frac{ak_{yb}}{4}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{q=1}^{Q}\left(A_{mn}\left(\frac{n\pi}{b}\right)-C_{mn}\right)\left(A_{mq}\left(\frac{q\pi}{b}\right)-C_{mq}\right)cos(n\pi)cos(q\pi)
\end{aligned} \tag{4.2.16}$$

and by collecting terms and simplifying one arrives at:

$$\begin{aligned}
U_{springs} = &\frac{b}{4}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{p=1}^{P}\left[\left(A_{mn}\left(\frac{m\pi}{a}\right)-B_{mn}\right)\left(A_{pn}\left(\frac{p\pi}{a}\right)-B_{pn}\right)\right]\left(k_{x0}+k_{xa}cos(m\pi)cos(p\pi)\right)\delta_{n,q}\\
&+\frac{a}{4}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{q=1}^{Q}\left[\left(A_{mn}\left(\frac{n\pi}{b}\right)-C_{mn}\right)\left(A_{mq}\left(\frac{q\pi}{b}\right)-C_{mq}\right)\right]\left(k_{y0}+k_{yb}cos(n\pi)cos(q\pi)\right)\delta_{m,p}
\end{aligned} \tag{4.2.17}$$

where $\delta_{n,q}$ and $\delta_{m,p}$ are the Kronecker delta which ensures that the integrals are only non-zero when p equals m and q equals n.

4.3 Stiffness matrices and eigenvalue equation

The eigenvalue problem to be solved is stated on the form:

$$\left(\mathbf{K}^M - \Lambda\mathbf{K}^G\right)\mathbf{A} = 0 \tag{4.3.1}$$

Where $\mathbf{K}^M$ is the material stiffness matrix, $\mathbf{K}^G$ is the geometric stiffness matrix, Λ is the eigenvalue and $\mathbf{A}$ the corresponding eigenvector. The material and geometric stiffness matrices are defined as:

$$\mathbf{K}^M = \begin{bmatrix} \mathbf{K}^M_{AA} & \mathbf{K}^M_{AB} & \mathbf{K}^M_{AC} \\ \mathbf{K}^M_{BA} & \mathbf{K}^M_{BB} & \mathbf{K}^M_{BC} \\ \mathbf{K}^M_{CA} & \mathbf{K}^M_{CB} & \mathbf{K}^M_{CC} \end{bmatrix} \quad \text{and} \quad \mathbf{K}^G = \begin{bmatrix} \mathbf{K}^G_{AA} & \mathbf{K}^G_{AB} & \mathbf{K}^G_{AC} \\ \mathbf{K}^G_{BA} & \mathbf{K}^G_{BB} & \mathbf{K}^G_{BC} \\ \mathbf{K}^G_{CA} & \mathbf{K}^G_{CB} & \mathbf{K}^G_{CC} \end{bmatrix}$$

Both the material and stiffness matrices have nine elements, but the size of the sub-matrices depends on how many terms in the series are needed for convergence. Each of the sub-matrices in the stiffness matrices are computed from the expression for potential energy:

$$\mathbf{K}^M = \begin{bmatrix} \dfrac{\partial^2 U}{\partial A_{ij}\partial A_{kl}} & \dfrac{\partial^2 U}{\partial A_{ij}\partial B_{kl}} & \dfrac{\partial^2 U}{\partial A_{ij}\partial C_{kl}} \\ \dfrac{\partial^2 U}{\partial B_{ij}\partial A_{kl}} & \dfrac{\partial^2 U}{\partial B_{ij}\partial B_{kl}} & \dfrac{\partial^2 U}{\partial B_{ij}\partial C_{kl}} \\ \dfrac{\partial^2 U}{\partial C_{ij}\partial A_{kl}} & \dfrac{\partial^2 U}{\partial C_{ij}\partial B_{kl}} & \dfrac{\partial^2 U}{\partial C_{ij}\partial C_{kl}} \end{bmatrix} \quad \text{and} \quad \mathbf{K}^G = \begin{bmatrix} \dfrac{\partial^2 W}{\partial A_{ij}\partial A_{kl}} & \dfrac{\partial^2 W}{\partial A_{ij}\partial B_{kl}} & \dfrac{\partial^2 W}{\partial A_{ij}\partial C_{kl}} \\ \dfrac{\partial^2 W}{\partial B_{ij}\partial A_{kl}} & \dfrac{\partial^2 W}{\partial B_{ij}\partial B_{kl}} & \dfrac{\partial^2 W}{\partial B_{ij}\partial C_{kl}} \\ \dfrac{\partial^2 W}{\partial C_{ij}\partial A_{kl}} & \dfrac{\partial^2 W}{\partial C_{ij}\partial B_{kl}} & \dfrac{\partial^2 W}{\partial C_{ij}\partial C_{kl}} \end{bmatrix}$$

However, because the only coefficient to appear in the expressions in W is A_{ij}, the only non-zero term in the geometric stiffness matrix is K^G_{AA}.

The terms appearing in the material and geometric stiffness matrices are all to be differentiated first with respect to coefficients in k and l and then with respect to coefficients in i and j. In the first derivation the chain rule is applied by first setting all m and n equal to k and l and differentiating with respect to these new coefficients and then setting all p and q equal to k and l and differentiating with respect to these new coefficients. The result after this first differentiation is an equation in m,n,p,q,k and l. Then the whole process is repeated when the result from the first derivation is again differentiated, but this time using the chain rule and replacing m and n with i and j and differentiating, and then p and q with i and j and differentiating. The result is finally an expression in i,j,k and l.

This process is shown on the first term, or sub-matrix, in the material stiffness matrix, the process is similar and therefore omitted for the other terms where only the results are shown. The first term is to be differentiated with respect to A_{ij}

and A_{kl}. Therefore it is only necessary to use the terms where A_{mn} appears twice:

$$\begin{aligned}\mathbf{K}^M_{AA} &= \frac{\partial^2}{\partial A_{ij}\partial A_{kl}}\left[\frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\left[D_x A^2_{mn}\left(\frac{m\pi}{a}\right)^4 + (D_x\nu_{yx}+D_y\nu_{xy})A^2_{mn}\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2\right.\right.\\ &\qquad\left.\left. + D_y A^2_{mn}\left(\frac{n\pi}{b}\right)^4 + 2D_{xy}A^2_{mn}\left(\frac{m\pi}{a}\right)^2\left(\frac{n\pi}{b}\right)^2\right]\frac{ab}{4}\right]\\ &= \frac{\partial}{\partial A_{ij}}\left[\frac{1}{2}\sum_{k=1}^{M}\sum_{l=1}^{N}\left[D_x 2A_{kl}\left(\frac{k\pi}{a}\right)^4 + (D_x\nu_{yx}+D_y\nu_{xy})2A_{kl}\left(\frac{k\pi}{a}\right)^2\left(\frac{l\pi}{b}\right)^2\right.\right.\\ &\qquad\left.\left. + D_y 2A_{kl}\left(\frac{l\pi}{b}\right)^4 + 4D_{xy}A_{kl}\left(\frac{k\pi}{a}\right)^2\left(\frac{l\pi}{b}\right)^2\right]\frac{ab}{4}\right]\\ &= \sum_{i=1}^{M}\sum_{j=1}^{N}\left[D_x\left(\frac{i\pi}{a}\right)^4 + (D_x\nu_{yx}+D_y\nu_{xy})\left(\frac{i\pi}{a}\right)^2\left(\frac{j\pi}{b}\right)^4\right.\\ &\qquad\left. + D_y\left(\frac{j\pi}{b}\right)^4 + 2D_{xy}\left(\frac{i\pi}{a}\right)\left(\frac{j\pi}{b}\right)\right]\frac{ab}{4}\end{aligned} \tag{4.3.2}$$

The rest of the terms (sub-matrices) in the material stiffness matrix are:

$$\mathbf{K}^M_{BB} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[2D_x\left(\frac{i\pi}{a}\right)^2 + D_{xy}\left(\frac{j\pi}{b}\right)^2 + 2S_x\right]\frac{ab}{4} \tag{4.3.3}$$

$$\mathbf{K}^M_{CC} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[2D_y\left(\frac{j\pi}{b}\right)^2 + D_{xy}\left(\frac{i\pi}{a}\right)^2 + 2S_y\right]\frac{ab}{4} \tag{4.3.4}$$

$$\mathbf{K}^M_{AB} = -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[2D_x\left(\frac{i\pi}{a}\right)^3 + (D_x\nu_{yx}+D_y\nu_{xy})\left(\frac{i\pi}{a}\right)\left(\frac{j\pi}{b}\right)^2 + 2D_{xy}\left(\frac{i\pi}{a}\right)\left(\frac{j\pi}{b}\right)^2\right]\frac{ab}{4} \tag{4.3.5}$$

$$\mathbf{K}^M_{AC} = -\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[2D_y\left(\frac{j\pi}{b}\right)^3 + (D_x\nu_{yx}+D_y\nu_{xy})\left(\frac{i\pi}{a}\right)^2\left(\frac{j\pi}{b}\right) + 2D_{xy}\left(\frac{i\pi}{a}\right)^2\left(\frac{j\pi}{b}\right)\right]\frac{ab}{4} \tag{4.3.6}$$

$$\mathbf{K}_{BC}^{M} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[(D_x\nu_{yx} + D_y\nu_{xy})\left(\frac{i\pi}{a}\right)\left(\frac{j\pi}{b}\right) + D_{xy}\left(\frac{i\pi}{a}\right)\left(\frac{j\pi}{b}\right)\right]\frac{ab}{4} \quad (4.3.7)$$

The only term in the geometric stiffness matrix is K_{AA}^{G}. This term also include the four series because of the integral requirements of N_{xy}:

$$\begin{aligned}\mathbf{K}_{AA}^{G} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[N_x\left(\frac{i\pi}{a}\right)^2 + N_y\left(\frac{j\pi}{b}\right)^2\right]\frac{ab}{4} \quad (4.3.8)\\ + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{P}\sum_{l=1}^{Q}\left[N_{xy}\left(\frac{i\pi}{a}\right)\left(\frac{l\pi}{b}\right)\left[\frac{a[1-(-1)^i(-1)^k]k}{\pi(k^2-i^2)}\right]\left[\frac{b[1-(-1)^j(-1)^l]j}{\pi(j^2-l^2)}\right]I_{ik}I_{jl}\right.\\ \left. + N_{xy}\left(\frac{k\pi}{a}\right)\left(\frac{j\pi}{b}\right)\left[\frac{a[1-(-1)^k(-1)^i]i}{\pi(i^2-k^2)}\right]\left[\frac{b[1-(-1)^l(-1)^j]l}{\pi(l^2-j^2)}\right]I_{ik}I_{jl}\right]\end{aligned}$$

The displacement vector contains the degrees of freedom and can be written as:

$$\mathbf{A} = \begin{bmatrix} A_{ij} \\ \vdots \\ A_{MN} \\ B_{ij} \\ \vdots \\ B_{MN} \\ C_{ij} \\ \vdots \\ C_{MN} \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{C} \end{bmatrix}$$

4.4 Contributions to stiffness matrix from elastic springs

When the elastic springs are added to the plate system, the material stiffness matrix is augmented by a stiffness matrix with terms from the elastic springs. The eigenvalue system to be solved is of similar form, but must now include this spring-stiffness matrix:

$$\left(\left(\mathbf{K}^{M} + \mathbf{K}^{S}\right) - \Lambda\mathbf{K}^{G}\right)\mathbf{A} = 0 \quad (4.4.1)$$

The terms in the spring stiffness matrix are found in the same way as the material and geometric stiffness matrices, by differentiation as described in Eq. (4.3.2).

The final terms are:

$$\mathbf{K}^S_{AA} = \frac{b}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{P}\left(\frac{k\pi}{a}\right)\left(\frac{i\pi}{a}\right)\left(k_{x0} + k_{xa}cos(k\pi)cos(1\pi)\right)\delta_{n,q} + \frac{a}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{l=1}^{Q}\left(\frac{l\pi}{b}\right)\left(\frac{j\pi}{b}\right)\left(k_{y0} + k_{yb}cos(l\pi)cos(j\pi)\right)\delta_{m,p} \tag{4.4.2}$$

$$\mathbf{K}^S_{BB} = \frac{b}{2}\sum_{i=1}^{M}\sum_{k=1}^{P}\left(k_{x0} + k_{xa}cos(i\pi)cos(k\pi)\right)\delta_{n,q} \tag{4.4.3}$$

$$\mathbf{K}^S_{CC} = \frac{a}{2}\sum_{j=1}^{N}\sum_{l=1}^{Q}\left(k_{y0} + k_{yb}cos(l\pi)cos(j\pi)\right)\delta_{m,p} \tag{4.4.4}$$

$$\mathbf{K}^S_{AB} = -\frac{b}{2}\sum_{i=1}^{M}\sum_{k=1}^{P}\left(\frac{i\pi}{a}\right)\left(k_{x0} + k_{xa}cos(i\pi)cos(k\pi)\right)\delta_{n,q} \tag{4.4.5}$$

$$\mathbf{K}^S_{AC} = -\frac{a}{2}\sum_{j=1}^{N}\sum_{l=1}^{Q}\left(\frac{j\pi}{b}\right)\left(k_{y0} + k_{yb}cos(l\pi)cos(j\pi)\right)\delta_{m,p} \tag{4.4.6}$$

4.5 Contribution to stiffness matrix from pre-stress

If a plate is already loaded up to some level when the buckling analysis begins, the eigenvalues will be lower. This can for instance be if a plate loaded in the transverse direction (normal to the longest edge) with a constant load and one wants to find how much load it can take in the longitudinal direction before buckling occurs. This pre-stress can easily be modelled in Rayleigh-Ritz by including the expressions for the load, not in the geometric stiffness matrix, which is linked with the eigenvalue, but subtracting it from the material stiffness matrix. Thus, just like the elastic springs on the boundary will raise the overall stiffness of the plate, the pre-stress loads will lower it. In mathematical terms this can be written as:

$$\left(\left(\mathbf{K}^M - \mathbf{K}^{pre}\right) - \Lambda\mathbf{K}^G\right)\mathbf{A} = 0 \tag{4.5.1}$$

where the terms in the pre-stress stiffness matrix are all, just like the terms in the geometric stiffness matrix, $\mathbf{K}^G$, terms in the coefficients $A_{ij}A_{kl}$. The terms depend on which load is used as pre-stress. Only one or two of the three $N1, N2$ and $N3$ is used as pre-stress at any time, while the other(s) are used in the buckling analysis. In this way interaction curves for pre-stressed plates can be plotted as well as those for initially unstressed plates. The only non-zero terms in the matrix are equal to those in the geometric stiffness matrix:

$$\begin{aligned}\mathbf{K}_{AA}^{pre} = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[N_{x,pre}\left(\frac{i\pi}{a}\right)^2 + N_{y,pre}\left(\frac{j\pi}{b}\right)^2\right]\frac{ab}{4} \qquad (4.5.2)\\ + \sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{P}\sum_{l=1}^{Q}\left[N_{xy,pre}\left(\frac{i\pi}{a}\right)\left(\frac{l\pi}{b}\right)\left[\frac{a[1-(-1)^i(-1)^k]k}{\pi(k^2-i^2)}\right]\left[\frac{b[1-(-1)^j(-1)^l]j}{\pi(j^2-l^2)}\right]I_{ik}I_{jl}\right.\\ \left. + N_{xy,pre}\left(\frac{k\pi}{a}\right)\left(\frac{j\pi}{b}\right)\left[\frac{a[1-(-1)^k(-1)^i]i}{\pi(i^2-k^2)}\right]\left[\frac{b[1-(-1)^l(-1)^j]l}{\pi(l^2-j^2)}\right]I_{ik}I_{jl}\right]\end{aligned}$$

where the appropriate pre-stress is set to a desired value and the two others are set equal to zero.

4.6 Static condensation of degrees of freedom

The Rayleigh-Ritz model of the sandwich plate has been implemented in FORTRAN as code. With a compiler program, the code is run and the eigenvalues are obtained. In order to make the code as fast as possible, the static condensation method has been used to make the stiffness matrix dimensions as small as possible. This can be done because the only term in the geometric stiffness matrix is $\mathbf{K}_{AA}^G$. The process is independent of the presence of both an elastic spring stiffness matrix and a pre-stress matrix, as these can both be included in the material stiffness matrix before static condensation is carried out. The system is now:

$$\left[\begin{bmatrix}\mathbf{K}_{AA}^M & \mathbf{K}_{AB}^M & \mathbf{K}_{AC}^M\\ \mathbf{K}_{BA}^M & \mathbf{K}_{BB}^M & \mathbf{K}_{BC}^M\\ \mathbf{K}_{CA}^M & \mathbf{K}_{CB}^M & \mathbf{K}_{CC}^M\end{bmatrix} - \Lambda\begin{bmatrix}\mathbf{K}_{AA}^G & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\right]\begin{bmatrix}\mathbf{A}\\ \mathbf{B}\\ \mathbf{C}\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}$$

From these three equations in $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, we see that only the first has anything to do with the external loads in $\mathbf{K}_{AA}^G$. This enables us to condense the system from a matrix of three by three sub-matrices, to a system of one sub matrix. From the last equation one can write out an expression for $\mathbf{C}$:

$$\mathbf{C} = -\mathbf{K}_{CC}^{M^{-1}}\left[\mathbf{K}_{CA}^M\mathbf{A} + \mathbf{K}_{CB}^M\right] \qquad (4.6.1)$$

This expression can now be used in the second equation to eliminate out $\mathbf{C}$ and obtain an expression of $\mathbf{B}$:

$$\mathbf{B} = -\left[\mathbf{K}_{BB}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]^{-1}\left[\mathbf{K}_{BA}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CA}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]\mathbf{A} \tag{4.6.2}$$

which can be used to eliminate out $\mathbf{B}$ from the first equation, and express this only in terms of $\mathbf{A}$:

$$\begin{aligned}&\Bigg[\left[\mathbf{K}_{AA}^{M} - \mathbf{K}_{AC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CA}^{M} - \Lambda\mathbf{K}_{AA}^{G}\right] \\ &- \left[\mathbf{K}_{AB}^{M} - \mathbf{K}_{AC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]\left[\mathbf{K}_{BB}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]^{-1}\left[\mathbf{K}_{BA}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CA}^{M}\right]\Bigg]\mathbf{A} = 0\end{aligned} \tag{4.6.3}$$

The condensed system can now be written on the form:

$$\left(\tilde{\mathbf{K}}^{M} - \Lambda\tilde{\mathbf{K}}^{G}\right)\mathbf{A} = 0 \tag{4.6.4}$$

where the condensed material stiffness matrix is:

$$\begin{aligned}\tilde{\mathbf{K}}^{M} = &\Bigg[\left[\mathbf{K}_{AA}^{M} - \mathbf{K}_{AC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CA}^{M}\right] \\ &- \left[\mathbf{K}_{AB}^{M} - \mathbf{K}_{AC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]\left[\mathbf{K}_{BB}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CB}^{M}\right]^{-1}\left[\mathbf{K}_{BA}^{M} - \mathbf{K}_{BC}^{M}\mathbf{K}_{CC}^{M^{-1}}\mathbf{K}_{CA}^{M}\right]\Bigg]\end{aligned} \tag{4.6.5}$$

and the condensed geometric stiffness matrix is:

$$\tilde{\mathbf{K}}^{G} = \mathbf{K}^{G} \tag{4.6.6}$$

4.7 Implementation of Rayleigh-Ritz in FORTRAN

The matrices derived in the previous section have been implemented in a FORTRAN computer program, and a eigenvalue solver has been used to solve for the eigenvalues. The model can be used for plates loaded in in-plane uni- or biaxial normal forces, in-plane shear forces, or any combination of these. The script includes the material and geometric stiffness matrixes for eigenvalue analyses of simply supported plates, and spring stiffness matrix which can be activated for analysis of clamped plates.

In reality a plate is almost always found to be elastically supported, that is a

place between simply supported and clamped, and because the boundary springs can be set to whatever value, this can be modelled too. In addition, because the springs on the boundary are independent of each other, these can be activated individually in order to model plates with some edges fixes and some edges simply supported. Free edges are for the moment not possible. A pre-stress matrix has also been programmed so that plates which are already under some load can be checked for buckling.

The FORTRAN script is shown in Appendix B.

Chapter 5

Modelling in ABAQUS CAE

5.1 Introduction

This chapter deals with the creation of the models in ABAQUS CAE (Complete Abaqus Environment)[1] and the challenges encountered in modelling different types of sandwich plates with this software. Before starting with the reference models, to which the results from the analytical models could be compared, a verification study was carried out to ensure correct results from these FEA models. A variety of elements were tested, mesh-refinements were carried out for the different elements and the models were checked for convergence. With the lessons learned from this small study, the next step was to create the Finite Element models for validation of the analytical and semi-analytical models developed in the previous chapters of the thesis.

There were also several problems encountered during the work of the thesis, some modelling techniques that worked well for some plates did not work at all for others. Some of the combinations of element type and mesh-size did not yield satisfactory results, some even failed completely. Others performed well, and from these analyses the 20-node solid brick element C3D20R and the 8-node shell element S8R have been chosen as the elements to use in the further analyses.

Throughout the thesis the notation of the plate has been $L1\text{x}L2\text{x}t1 - tc - t2\text{mm}$, where $L1$ and $L2$ are the in-plane dimensions and $t1$, $t2$ and tc denote the thickness of the plate layers as defined in the notation list.

[1] ABAQUS is a Finite Element software created by Simulia www.simulia.com

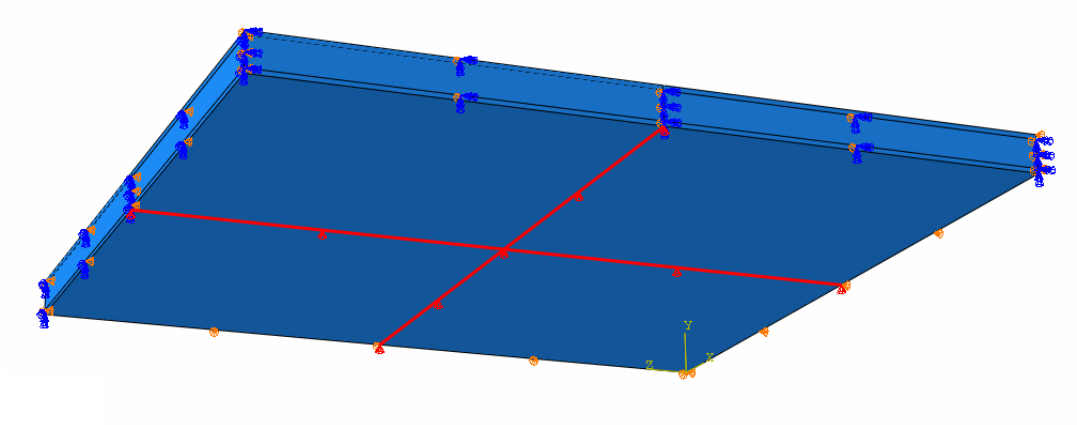


Figure 5.1: ABAQUS solid element model with the out-of-plane boundary conditions along the two red intersecting lines in the bottom of the plate. In this way, the edges are free to move out-of-plane, while the two lines are fixed in this direction.

5.2 Modelling

5.2.1 Different element types

In Finite Element analyses, plates can be modelled either as a three-dimensional solid or as a two-dimensional shell. For an isotropic plate the difference in modelling with solids or shells will probably be negligible as the proper shell elements can accurately describe the behaviour exhibited by a plate in bending or buckling. For the type of sandwich considered in this thesis, however, an often considerable part of the deflections will be the transverse shear deformations in the core due to the low shear stiffness of this layer. This means that the cross-sections will be linearly continuous only within each layer, but have a change of slope between the core and face layers. This behaviour will not be caught either by the analytical formulae presented in Chapter 2, the Rayleigh-Ritz model presented in Chapter 4 or the shell elements in ABAQUS. These theories all include the transverse shear deformations because of the low shear stiffness of the core, but they also assume that this shear deformation is linear and continuous over the entire cross-section. Modelling with solid elements, however, will make this effect visible.

5.2.2 Solid elements

The plates that were modelled with solid elements were all made up geometrically of three distinct layers, with different material properties assigned to each layer. The three layers were assembled into one structure. All plates were meshed with respect to element size, type and number of elements according to the conclusions

found in the convergence study described below. For pressure analyses the load was simply applied as a uniform pressure load to one of the face plates, and the plate was laid up as a simply supported plate at the lower edge of the lower face plate. For clamped plates it was very easy to perform the buckling analyses as the edges were fixed, but it quickly became apparent that this was not as easy when it came to the buckling analyses of simply supported plates.

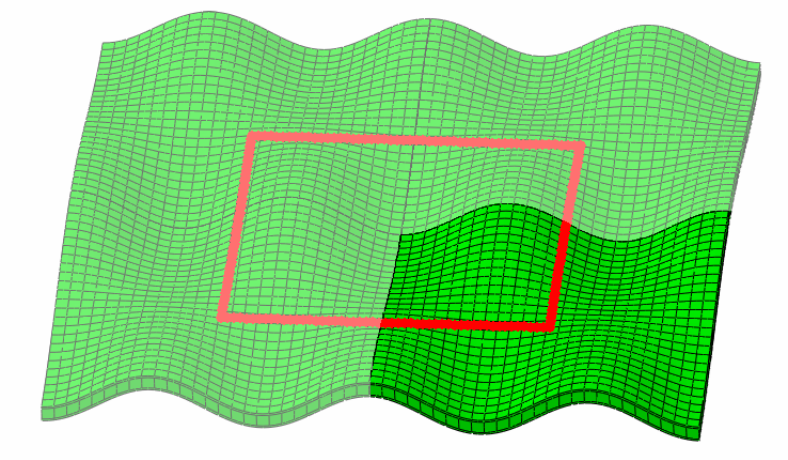


Figure 5.2: Buckling shape of solid model plate in ABAQUS. The dark green quarter shows the modelled plate, while the red lines show the boundaries of the theoretical plate. Symmetry has been used to achieve this.

After some problems, which are described in Section 5.3, the following method was found to be the most accurate. Two adjacent edges were fixed for rotation and in-plane displacement and were only free to move out-of-plane. On the two opposing edges, symmetry conditions were used. The out-of-plane boundary condition was specified along two intersecting lines on the bottom of the lower face plate, see Fig. 5.1.

The theory behind this approach was that by using symmetry conditions on the two edges not used for in-plane support, one fourth of the *modelled* plate would represent the *theoretical* plate (see Fig. 5.2). This ensured that the edges of the *modelled* plate remained free of rotations, making it easier to correctly introduce the in-plane loads to the structure. As the edges with symmetry conditions of the *modelled* plate would represent the middle of the *theoretical* plate this would also be correct. The reason is that the rotation of planes parallel with the edges of a plate should at the middle of the plate, $(a/2, b/2)$, remain zero during deformation (this assumes that the plate buckles into an odd number of half-waves and is symmetrically loaded by normal forces). In the same manner, the edges at

the centre of the *modelled* plate would rotate freely, thus representing the edges of the boundary of the theoretical simply supported plate, shown in red in the figure. It is important to point out that this approach will only work for buckling modes with an odd number of half-waves. Another limitation of solid modelling of sandwich plates is that in-plane shear loading proved to be so difficult to model correctly that it was abandoned completely and these loading scenarios were successfully modelled with shell elements instead. The approach described above was nonetheless perfect for visually showing the effect of shear deformations through the layers of the plate in uniaxial buckling and pressure loading, see Fig. 5.3.

Partial deflections

The partial deflection theory laid out in Section 2.4 says that the total transverse deflection w can be split into two parts - w_b from bending and w_s from transverse shear. To verify this theory, some of the solid element models were first allowed to deform as they should when the effect of transverse shear is included. Then a constraint was enforced on the plate, which specified that all nodes that initially shared x- and y-coordinates through the plate should remain on a straight line during the analysis. In essence this is the same as enforcing Navier's hypothesis and thereby stiffening the plate against transverse shear deformations.

By using the constraint called *slider* on all nodes through the thickness of the plate (sharing x- and y-coordinates) this was easily achieved in ABAQUS. This constraint specifies two reference nodes and forces any slave nodes to lie on a straight line created between the two master nodes. In the models in question, the two reference nodes were always the bottom node of the lower face plate and the corresponding top node in the upper face plate. Because ABAQUS CAE does not support the slider constraint in the graphical interface, a Python script had to be developed. This script is shown in Appendix C.1. After running a normal analysis without any sliders, the script was used to loop through the coordinate list in the output files. For each node with z-coordinate equal to zero (bottom node of lower face plate) the corresponding top node on the upper face plate was found, these were the two reference nodes. Then the script looped through all the nodes not on the top or the bottom of the plate and if the in-plane coordinates were the same as a set of top and bottom nodes, a slider MPC command was written to a text file; one line for each intermediate node. The lines written with the script were then copied into the model via the *Edit keyword* option in ABAQUS CAE. The analysis was then run one more time, and the plate was thus forced to disregard transverse shear deformations in the core.

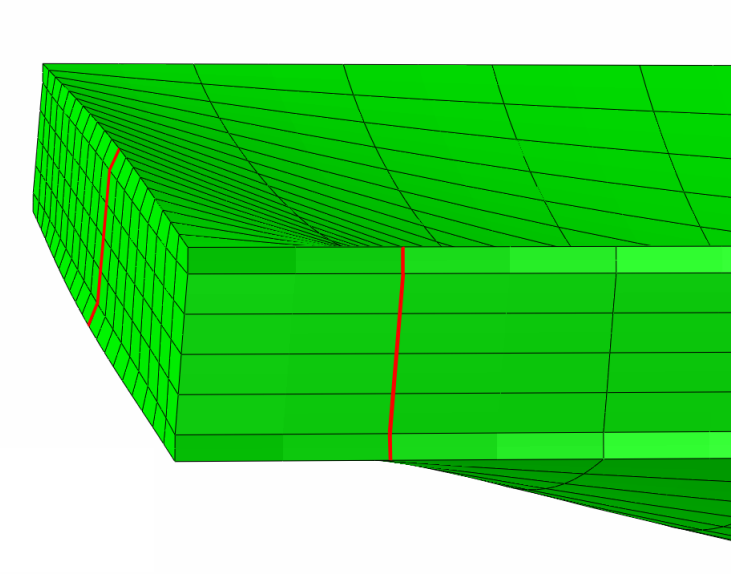


Figure 5.3: Screen capture from ABAQUS shows the edge of a deformed solid model plate. The red lines show the change of slope in the different layers of the sandwich plate. This effect can only be observed by using solid elements in ABAQUS.

5.2.3 Shell elements

Shell element models were created by applying a composite lay-up to a two-dimensional rectangular part. The material properties of each layer and the layer thicknesses were entered into the composite lay-up manager and from that the program created an equivalent stiffness, much like what is done in the analytical and Rayleigh-Ritz approaches in this thesis, and applied that to the plate. One major disadvantage over the solid element models was, of course, that there was no longer any way to visually observe the cross-sectional rotations, and especially the variation in slope between the core and the face layers. The advantages, however, of using shell elements included faster modelling time, possibility of in-plane shear (and combined) loading, accurate comparison between FEA and Rayleigh-Ritz and analytical solutions, more control over boundary conditions and easier loading of the model. It was apparent though, that the shell elements did not perform well in every situation either, as described in Section 5.3.

5.3 Problems encountered in modelling

5.3.1 Problems with solid model plate

In order for the buckling analyses of the simply supported plates to be accurate, the load would have to be introduced into the plate as correctly as possible and the edges of the plate would have to be able to deform as they wanted, with no constraints. The most obvious thing to do would be to use nodes along the middle plane edges of the plate for both in-plane and out-of-plane support and apply the load to the steel face plates. When the analyses were run, however, the soft material of the core made the supports move into the plate (or the plate warp around the supports), see Fig. 5.4. One solution to this problem was to use the *slider* constraint on the nodes along the edges of the core. Now the master nodes were not the top and bottom nodes of the plate, but the nodes in the interface between the core and face layers. In this way the core would remain straight during deformation and the observed change in rotation between the core and face layers would be preserved. This fixed the warping effect, but as the slider constraint does not specify the *distance* between the nodes a new problem arose instead. In some cases the effect was now that the plate collapsed onto itself instead. This could have been fixed by another constraint in ABAQUS called *link*, but it was not found to be possible to use both sliders and links at the same time.

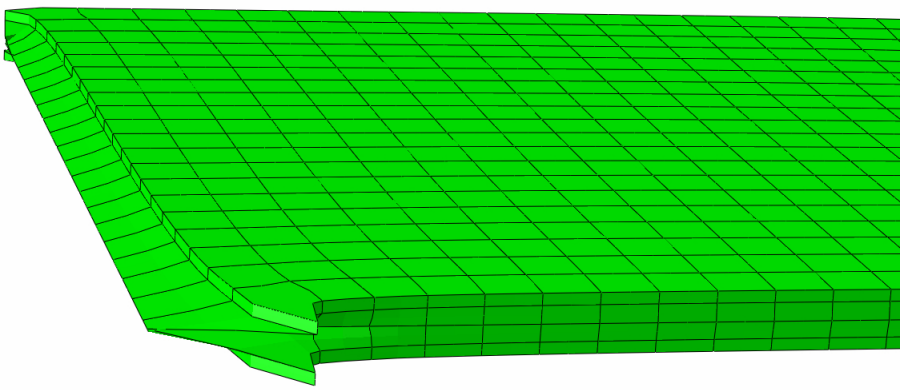


Figure 5.4: Screen capture from ABAQUS shows the edge of a deformed solid model plate. Here the core has stretched because the forces were applied in the steel face plates and the boundary conditions in the relatively soft core. This erroneous buckling shape was one of several problems encountered with solid models.

5.3.2 Problems with shell model plate

When the shell elements models were created, they were first compared to PULS [2] using steel in the core layer. For the simply supported plate, the correct values were only found when the nodal degrees of freedom on the boundary were fixed against rotations in the two directions normal to the edge. In most cases this worked well, however in some cases there seemed to be some wrinkling (individual face layer buckling), thus giving both erroneous buckling loads and shapes. When the two nodal rotation degrees of freedom were loosened, the wrinkling problem disappeared but another problem could then occur. The number of half-waves in the buckling shape was not the same as found with the Rayleigh-Ritz method or the analytical formulae. When the plate was made bigger, this problem would disappear, leading to the suspicion that there was some kind of limit in ABAQUS where the type of isotropic sandwich plates considered here cannot be accurately modelled. Looking back at Fig. 2.6 one can see that the buckling mode is very sensitive to changes as the shear factor increases, and as $\pi^2\theta$ reaches 0.5 and above, it is difficult to even distinguish the modes from each other.

Another problem that was encountered was when looking at a large plate of 4190x2800mmx4-25-4, the same model would not work for a plate of 1600x800x5-30-5mm. An eigenvalue would be found, but the buckling shape was completely without relevance. This was only the case when the material properties were that of the sandwich plate, not when changing the core layer to steel, meaning that there exists some kind of limit in ABAQUS also for the slenderness of the plate or ratio of stiffnesses. The problems were not investigated further in this thesis.

5.4 Convergence analysis

5.4.1 Analysis parameters

In order to verify that the analytical models gave accurate results they had to be compared to something that was known to be correct. ABAQUS was used to create these verification models. However, the accuracy of any Finite Element model depends on several factors, including the type of element and the size of the elements in the mesh. It cannot be known whether a model gives the correct results before it has been checked for convergence, i.e. seeing if more elements in the mesh leads to better results. If the model yields the same results after mesh refinement the model has converged. Here lies the importance of this study.

[2] PULS - *Panel Ultimate Limit Strength* is a semi-analytical elastic buckling and ultimate strength program for stiffened and unstiffened isotropic plates, developed at DNV.

	Steel	Polyurethane
Young's modulus - E	208000 MPa	750MPa
Shear modulus - G	80000 MPa	288 MPa
Poisson's ratio - ν	0,3	0,3
Yield stress[a]- σ_F	235 MPa	N/A
Strain hardening parameter[a]- E_T	1000 MPa	N/A

[a] Only relevant for elasto-plastic and ultimate capacity analyses

Table 5.1: Material properties of core and face sheets used in Finite Element models and analytical models.

The material properties as shown in Table 5.1. The size of the plate was set to 800x800mm and created with face plate thickness of 5mm and core thickness of 30mm for a total plate thickness of 40mm. Section properties were assigned to each of the three layers. At first, the core was also modelled with steel material in order to verify the model against the program PULS. When agreement was obtained, the core material was changed to polyurethane.

The analyses were started with one element in each layer (one in each face plate and one in the core) and if the results were not satisfactory in either eigenvalue, buckling shape or both, the number of elements through each layer of the model was increased. In the same way, all models were started with a relatively coarse mesh of 50x50mm (in the plane of the plate) and the mesh refined to 40x40mm, 30x30mm and 20x20mm, and in some instances even 15x15mm and 10x10mm. Not all models were refined through all the levels of element sizes, especially the second order elements C3D20R gave satisfactory results already with the one element in each layer and the largest element size (50x50mm). For this element type, mesh refinement did not yield any improvement.

5.4.2 Results - solid element model

The plates in the solid model convergence analysis were modelled with the following elements:

- C3D20R - A 20-node brick element with quadratic interpolation and reduced integration
- C3D8R - An 8-node brick element with linear interpolation and reduced integration

- C3D8I - An 8-node brick element with linear interpolation and incompatible modes

For the 800x800x5-30-5mm plate, the quadratic 20-node solids gave the correct results already at 50x50mm element size and one element in each layer. This was confirmed by refining the mesh and obtaining the same eigenvalue and buckling shape and by comparing an all-steel plate model with results from PULS. The linear 8-node solid element with reduced integration C3D8R did not give valid buckling shapes until three elements were used in each layer and an element size of 20x20mm. Even then, the computed eigenvalue was 12% higher than that computed from the quadratic element C3D20R, and it was not until the element size was reduced to 10x10mm that the results from the two element types were within 1% of each other. This requirement of extremely many elements in the model had a significant impact on the computational time, in fact the model with 10x10mm C3D8R elements and three elements in each layer took almost three hours to compute, the 50x50mm C3D20R with one element in each layer took under five minutes.

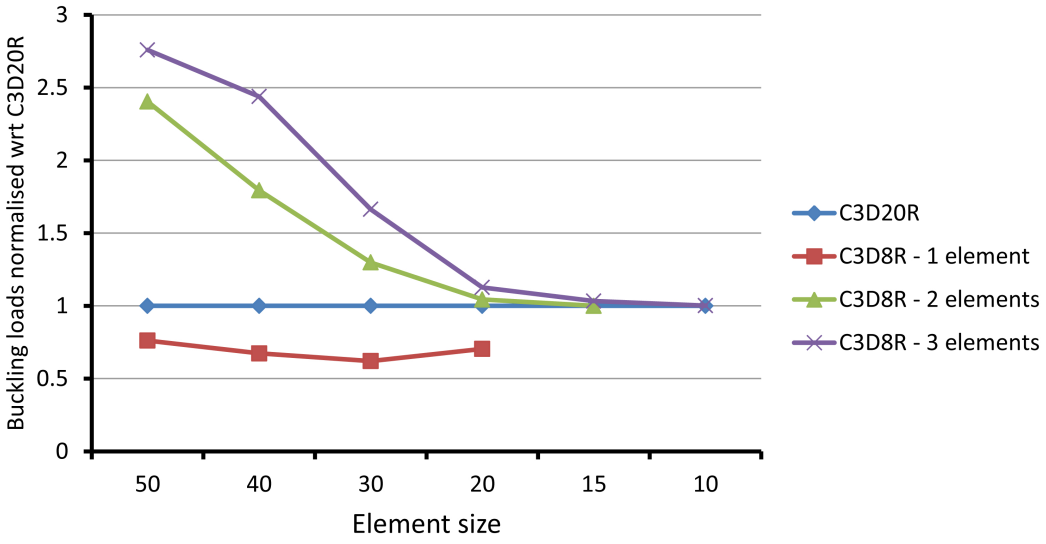


Figure 5.5: Eigenvalues computed for an 800x800x5-30-5mm plate in ABAQUS with different solid element types and sizes. The eigenvalues have been normalised with respect to the one found with the quadratic element C3D20R at 50x50mm element size, which had converged.

In fully integrated first order elements, the interpolation functions make the element unable to describe curved surfaces [12]. Hence, during bending, these ele-

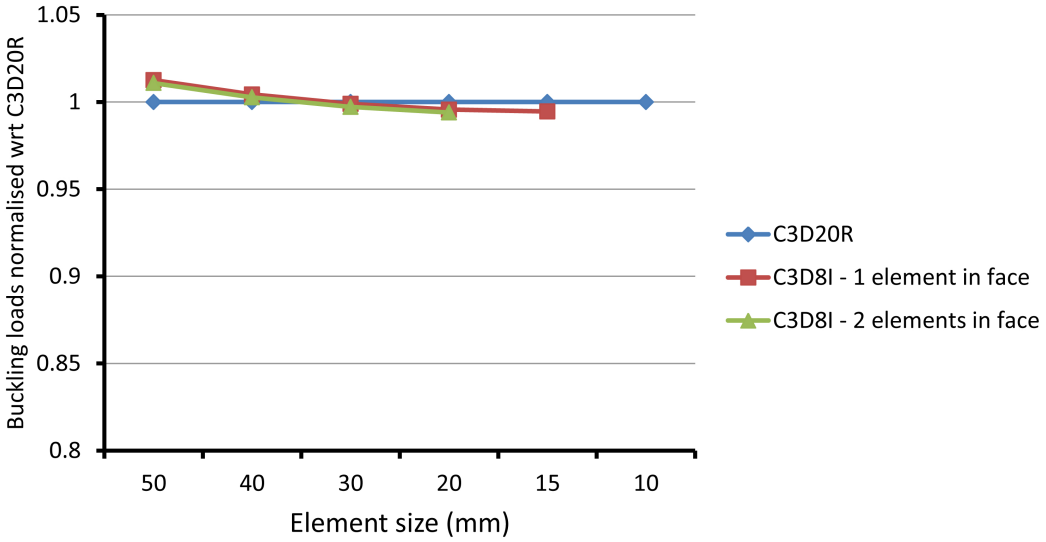


Figure 5.6: Eigenvalues computed for a 1600x800x5-30-5mm plate in ABAQUS with solid element C3D20R and linear element C3D8I with reduced integration. The eigenvalues have been normalised with respect to the one found with the quadratic element C3D20R at 50x50mm element size, which had converged.

ments distort in a shearing rather than bending mode, and these artificial shear deformations absorb strain energy making the element too stiff. This effect is known as shear locking or parasitic shear. To overcome this, elements with reduced integration have been developed. While the second-order elements, such as C3D20R and S8R performed well in most cases, the first-order elements such as C3D8R and S4R suffered from another problem known as hourglassing. While the fully integrated elements use several integration points within each element, the reduced integration elements use very few. In fact, the fully integrated C3D8 element uses 8 points, while the corresponding C3D8R (reduced) uses only one point. This is where the problem arises; with only one integration point, the element can deform in such a way that the strain energy calculated at the single integration point equals zero. Hence, like the shear locking made elements too stiff for fully integrated elements, hourglassing makes the reduced integration elements too flexible. This can be overcome by using a fine mesh according to the ABAQUS manual [15], but as seen above the computational time is very high.

Another type of element was also tested, a first order element with incompatible modes, C3D8I. This element has been created to combat the parasitic shear by introducing incompatible modes to improve bending behaviour. As can be seen in Fig. 5.6, these elements perform, at least for the test plate, almost as well as the

second order C3D20R element. Already at 50x50mm element size and one element in each layer, this element was able to yield an eigenvalue within 2% of that computed with quadratic element C3D20R. The second-order elements with reduced integration do not suffer from hourglassing and is the best choice for modelling bending behaviour [15] and [16].

5.4.3 Results - shell element model

The shell models were created with the same geometry and material parameters as above, and two different elements were tested. These were:

- S8R - An 8-node shell element with reduced integration
- S4R - A 4-node shell element with reduced integration

The difference in the two elements was dramatically smaller than for the solid elements. The quadratic element S8R with 50x50mm element size was used as a baseline when it was clear that it had already converged. The model with the linear S4R element and 50x50mm size yielded a buckling load only 0,3 percent over that of the corresponding quadratic S8R element, see Fig. 5.7. This was something completely different to the solid models. The S4R models were tried out with progressively smaller element sizes, down to 20x20mm. This was done to see if the eigenvalue had indeed converged already at 50x50mm, or if it was bound for some lower level than the quadratic element.

Another interesting point was that for all the analyses, the shell element models yielded lower eigenvalues than the equivalent solid element models. This was somewhat unexpected, as it was thought that introducing more degrees of freedom into the model, as is done with solid elements, would make it softer. Here, of course, the difficulties in modelling exact boundary conditions for the solid model might play a significant role.

5.4.4 Conclusions

Based on the findings of this convergence study and references [15], [16] and [17], the second order elements with reduced integration C3D20R and S8R were chosen for use in further ABAQUS analyses in this thesis. Shell elements have been used for the most part for the analyses, except where the effect of transverse shear deformation was to be visually verified. This leads to simpler models, easier boundary conditions and easier loading, especially loading with in-plane shear, than for solid element models.

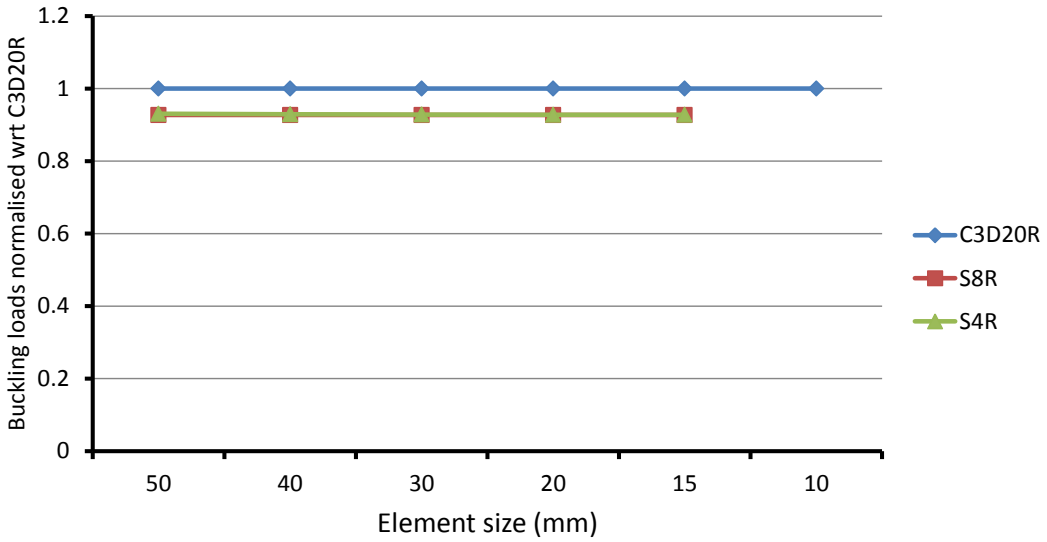


Figure 5.7: Eigenvalues computed with shell elements S4R and S8R and normalised with respect to solid element C3D20R. Both shell element models yield lower eigenvalues than the solid element model, but the difference between the linear S4R and the quadratic S8R elements is minute.

Chapter 6

Sandwich plate calculator

6.1 Introduction

A classification note for Steel Sandwich Panel Construction, CN.30.11[1], has been developed at DNV simultaneously with this thesis. The buckling part of the classification note is, like this thesis, based upon the analytical closed form formulae found in Zenkert [3], so extensive testing has been carried out in verifying the theory. The best way of doing this was in comparing the sandwich theory presented in Chapter 2 with ABAQUS for a variety of different plates. In this process, Excel has been used to make a calculator; a Visual Basic-automated spread-sheet. This calculator accepts input such as material properties, loads and plate geometry, and outputs this either as mid-plane deflection (if the load is lateral pressure) or a set of eigenvalues (if the loads are in the plane of the plate). A combined loading of lateral pressure and in-plane loads is currently not supported, although the separate lateral pressure and buckling analyses can be run at the same time. The results have been compared with both ABAQUS, for sandwich plates and all-steel plates, and with PULS for all-steel plates.

6.2 SSPC calculator

A spread-sheet called *SSPC calculator* was made in Excel, see Fig. 6.1. The spread-sheet and the classification note do not take into account the interaction between lateral pressure and buckling loads, i.e. both analyses can be computed simultaneously, but the results from the either does not affect the other.

The main concerns in designing the user interface were simplicity and clarity. The column to the left are all input cells. These cells are arranged in groups of geometric input, material input and input loads and allowable usage factors. The

column to the right include all the output values grouped by stiffness coefficients (that are displayed in both types of analyses), buckling output (only displayed if in-plane loads are present) and pressure output (only displayed if lateral pressure is present). Different colours have been used in the columns, yellow indicate input, while blue indicate output. The two shades of each colour were chosen to make it easier to distinguish between the lines in the spread-sheet. The layout has changed somewhat over the developing months, but it basically remains the same as when first started.

By pressing the button called *Calculate panel capacity*, a sequence of Visual Basic code will initiate the background calculations that will quickly be displayed in the output cells. The code is made up of a number of smaller scripts, some of which are common for all the plate analyses and some of which are only called in the event of buckling or pressure. Built into the code are controls that ensure that certain material property limits are not exceeded, and controls that determine if the problem is one of buckling, pressure or both, depending on which input cells are filled. Various pop-up warnings have been built in; some as messages with an option of yes or no (such as telling that the core stiffness is higher or lower than some limits and asking yes/no for continued calculations), some as pure messages that say that the calculations have been halted (such as a core layer Young's modulus higher than that of the face plate material). A plastic yield check for the face plate material has also been included, that will halt the program and display a pop-up message. If everything is ok, no pop-up messages should be presented to the user, and the output is displayed in the right column. Also included in the calculator are a check of core shear stress fracture and interface bonding shear stress. These two factors are part of another on-going work at DNV and has not been considered in this thesis. It has nonetheless been included in the calculator.

The values of capacity utilization are displayed in the top centre. These centre cells display the capacity values much larger and are meant to stand out as a clear indicator whether or not the plate being considered will be OK under the current configuration. In addition to the numerical values, the background colour of these cells will assist in determining the plate capacity. If the value is lower than that defined by the *allowable usage factor* input, the background will turn green. If not, it will turn red. If the value is not available, such as pressure capacity utilization in a pure in-plane load scenario, the background colour is yellow and "N/A" will be displayed in lieu of a numerical value. Because the pressure will give a known deflection value, this value can be used to find the maximum stress in the face plates, the shear stress in the core and the shear stress in the interface. Based upon this, the one actual usage factor closest to its corresponding allowable

usage factor will be used to calculate the maximum plate pressure. This will be displayed just below the three usage factors, and a black arrow will point to the one of the three usage factors that was the most critical.

A screen shot of the spread-sheet is shown in figure 6.1.

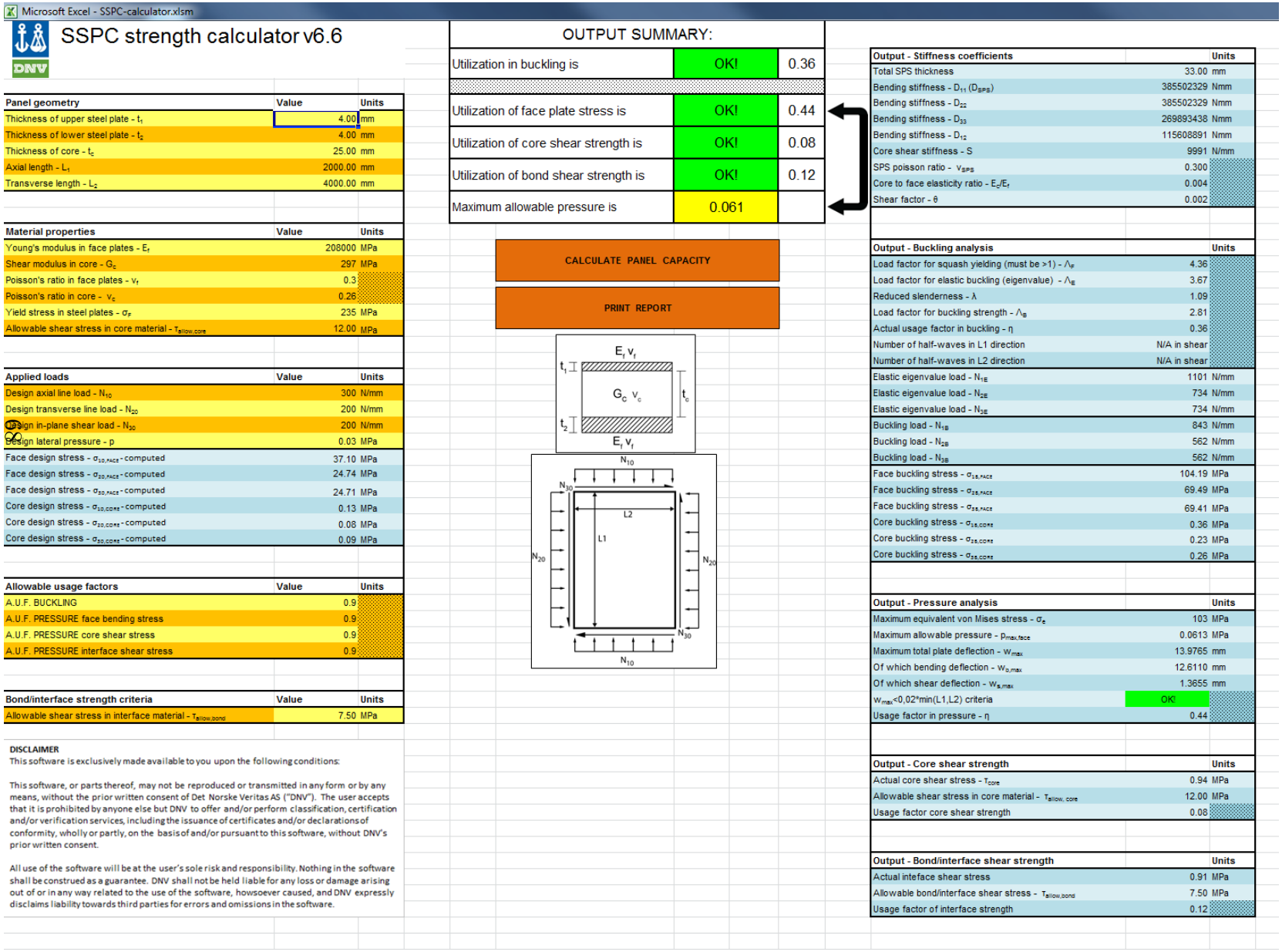


Figure 6.1: Screenshot of the SSPC calculator in Excel

Chapter 7

Results

7.1 Introduction

In this chapter the results from the verification of the various models are presented. Both the analytical closed form solutions presented in Chapter 2 and the Rayleigh-Ritz model presented in Chapter 4 have been compared to Finite Element models in ABAQUS. For the lateral pressure analyses, simply supported plates both analytical and with ABAQUS solid elements are compared. For the buckling analyses, analytical simply supported and Rayleigh-Ritz simply supported and clamped models are compared to ABAQUS shell elements. The last part is a comparison between the simply supported analytical elasto-plastic model and an ABAQUS ultimate capacity model using shell models.

7.2 Lateral pressure analysis

7.2.1 Introduction

A plate subjected to lateral pressure will deflect. In sandwich plates this deflection will be the sum of two deformations, bending and transverse shear, as described in Section 2.4 and computed by Eqs. (2.6.6) to (2.6.8). Failing to include the effects of transverse shear deformations might for some plates severely under-predict the lateral deflections, whereas for other plates it might not play a significant role at all. It is known that for an isotropic steel plate, the effects of the transverse shear become more important as the plate length to thickness ratio decreases to around 10 [4], and that these effects will be less important as the ratio increases. It is therefore reasonable to assume the same for sandwich plates. That is, for some plates the shear deformations will be of great importance to the overall deformations, whereas for other plates the shear deformations will be of little

importance. For sandwich plates, however, it is not only the ratio of length over plate thickness that will be governing, but also the thickness of the face sheets and the thickness of the core.

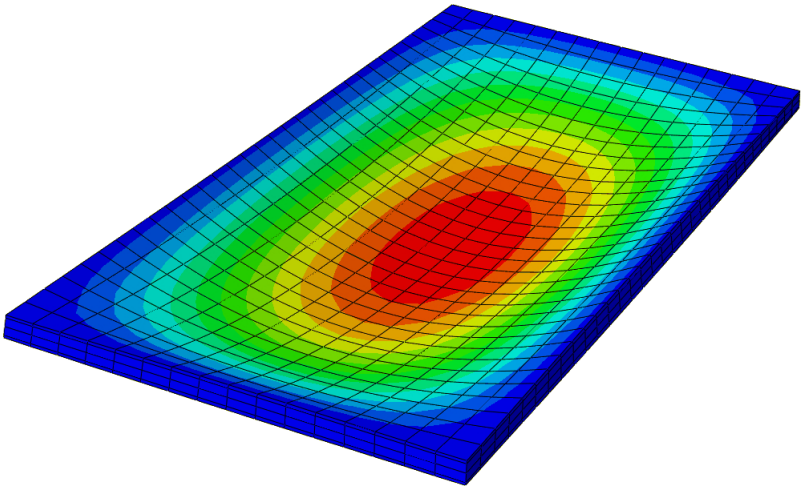


Figure 7.1: ABAQUS solid element model showing deflection due to lateral pressure on a simply supported plate.

7.2.2 Effect on transverse shear deformations of face to core thickness ratio

Using Eqs. (2.6.6) to (2.6.8) the total deflection w, deflection due to bending w_b, and deflection due to shear w_s, have been plotted for two square plates. One plate was 2000mm square and the other was 500 square. Both plates had a total thickness of 30mm, but the face and core thicknesses were varied. The boundary conditions for these plates were simply supported along all four edges and the applied load was 0.03MPa.

In the larger of the two plates the bending deformation is dominating the total deflection, as one can see from Fig. 7.2. As the core gets thicker and the face sheets thinner, one can see that the total deflection increases but the shear deformation are more or less constant. So changing the ratio of the core to face sheet thickness has no apparent effect on the shear deformations other than making it a smaller part of the total deformation, as long as the in-plane dimensions and the total plate thickness remains the same. This can be explained by the expressions for the bending and shear deformations, Eqs. (2.6.7) and (2.6.8), respectively. The equation for bending deflection is inversely proportional to the flexural rigidity, given by Eq. (2.5.4). This equation is itself proportional to both the face and

the core thicknesses, but also to the Young's moduli of the two materials. As the core thickness increases, the third term in Eq. (2.5.4) also increases, but because the relative insignificance of this term compared to the first two, which are coupled with a Young's modulus almost 300 times larger, the total flexural rigidity decreases.

The equation for shear deflection is inversely proportional to the shear stiffness, given by Eq. (2.5.5). This is only dependent on the shear modulus of the core, not the Young's moduli of the two materials. So as the face thickness decreases, this has no apparent effect on the shear stiffness, and hence the shear deformation part of the total deformations.

In the smaller plate, the shear deformations are still more or less constant, but they are, for all the same configurations of core and face sheet thicknesses, larger than the bending deformations. Seeing as the w_b-term is equal to Kirchhoff thin plate theory, shear deformation theory becomes more and more important as the sandwich plate gets stockier and w_s becomes more important.

The results have been compared to solid element models in ABAQUS (example model in Fig. 7.1) and as one can see from the two graphs, Figs. 7.2 and 7.3, there is good agreement between the analytical and Finite Element models. It is perhaps the stockiest of the two plates (500mm) that has the best agreement overall. The thinner plate (2000mm) has excellent agreement between the analytical and Finite Element models for the bending deformation, while the shear deformation is larger in the Finite Element model. For both cases the analytical models yield a bit smaller total deflection than the Finite Element models, this might be due to small differences in the boundary conditions between an analytical closed form solution and a solid element model with several layers. For all practical purposes the analytical formulae can be said to be in good agreement with Finite Element analyses. It is also to be noted that the analytical closed form model might have yielded better results compared to shell element models as these two theories are more similar, but as described in Chapter 5 there was no method of enforcing Navier's hypothesis in shell elements like the sliders could do with the solid elements. The partition of the total deflection into bending and shear parts is thus not possible with shell element models.

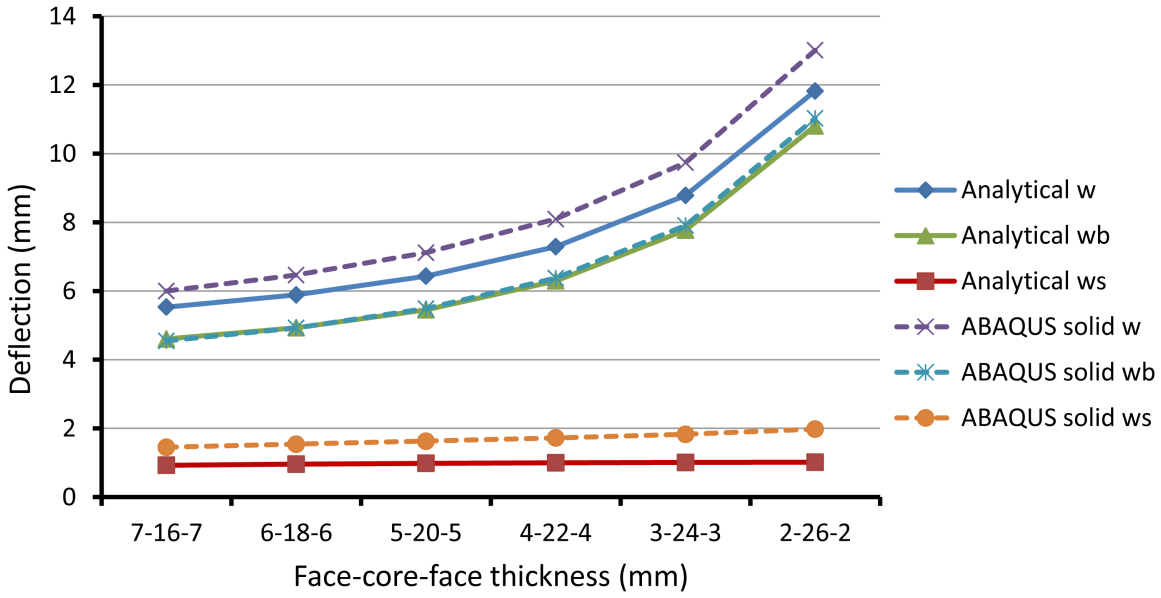


Figure 7.2: Deflection of a square plate with sides of 2000mm and total thickness of 30mm. Different ratios of core and face sheet thicknesses along the x-axis.

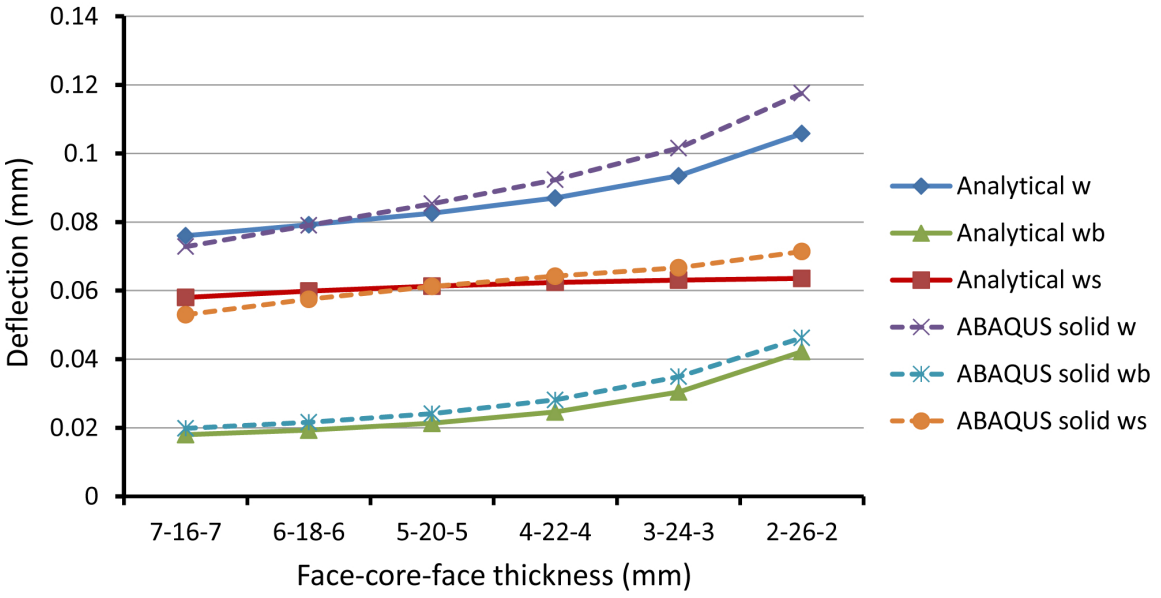


Figure 7.3: Deflection of a square plate with sides of 500mm and total thickness of 30mm. Different ratios of core and face sheet thicknesses along the x-axis.

7.2.3 Effect on transverse shear deformations of varying face sheet thickness

To see the effect on shear deformations by varying the face thickness, 16 different simply supported plates were checked for lateral deflections. The plates were all square, with dimensions from 500mm to 4000mm, with increments of 500mm. For 8 of the plates the core thickness was set to 10mm and for the other 8 plates the core thickness was set to 50mm. For each of the 16 plates, the face sheets were varied between 2mm and 5mm with 1mm increments. The applied pressure was again 0.03MPa.

Fig. 7.4 shows the shear deformation part of the total deformations for 4 different face thicknesses with the core constant at 10mm. Each core-face combination has been checked for the 8 different in-plane dimensions described above. Fig. 7.5 shows the same for the plates with a 50mm core thickness and varying face thickness, checked at the same 8 in-plane dimensions from 500mm to 2000mm.

For all cases the shear part of the total deflection decreases as the plate gets bigger in the plane, and also as the face thickness is increased. With increased face thickness comes increased flexural rigidity of the cross-section, and the plate is able to carry more of the total deflection through bending of the faces, hence the shear deformation part decreases. This can be explained in the same way as in the previous section. The bending deflection part is dependent on the flexural rigidity which is itself heavily influenced by the thickness of the faces, but not the core. The shear deflection part is inversely on the shear stiffness, which is not very sensitive to variations in the layer thicknesses.

Another thing one can see from comparing Figs. 7.4 and 7.5, is that for the plates with a thick core at 50mm, the shear deformation w_s is for all plates a larger part of the total deformation w than in the case of the thinner plates with core thickness of 10mm. The reason for this might be that the thicker the core gets when keeping the face thicknesses the same, the more the face plates will be able to slide relative to each other and the transverse shear deformation will be a larger part of the same deformations. Another effect is that as the core thickness is increased, the flexural rigidity of the plate increases. This makes the plate stiffer in bending, and the bending part of the total deflection will be smaller.

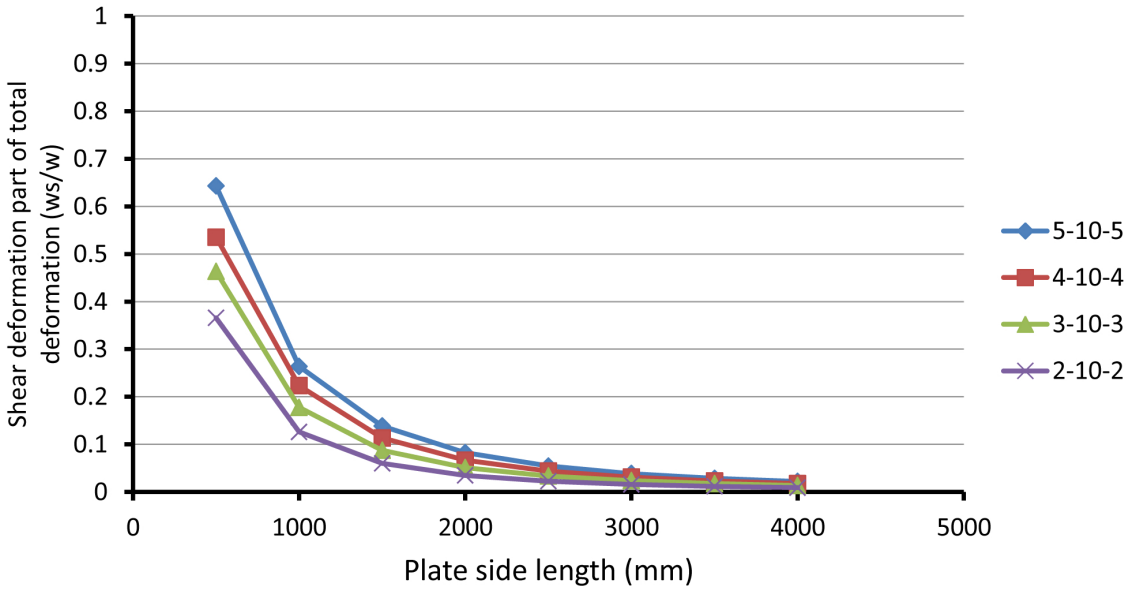


Figure 7.4: Shear deformation as part of total deformation for plates with constant core thickness of 10mm and varying face thickness between 2mm and 5mm.

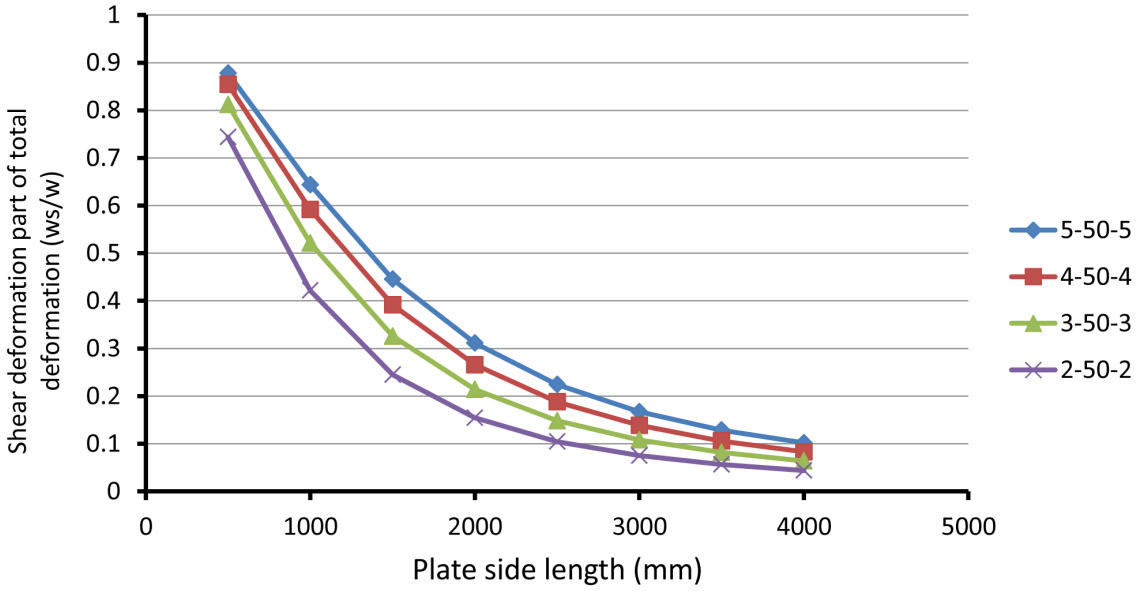


Figure 7.5: Shear deformation as part of total deformation for plates with constant core thickness of 50mm and varying face thickness between 2mm and 5mm.

7.2.4 Effect on transverse shear deformations of varying the core thickness

To see the effect on shear deformations by varying the core thickness, a new set of 16 plates was considered. The in-plane dimensions were still from 500mm to 4000mm with increments of 500mm. Eight of the plates had face thickness of 2mm and eight had a face thickness of 5mm. The core was varied from 10mm to 50mm with 10mm increments for each of the sixteen plates.

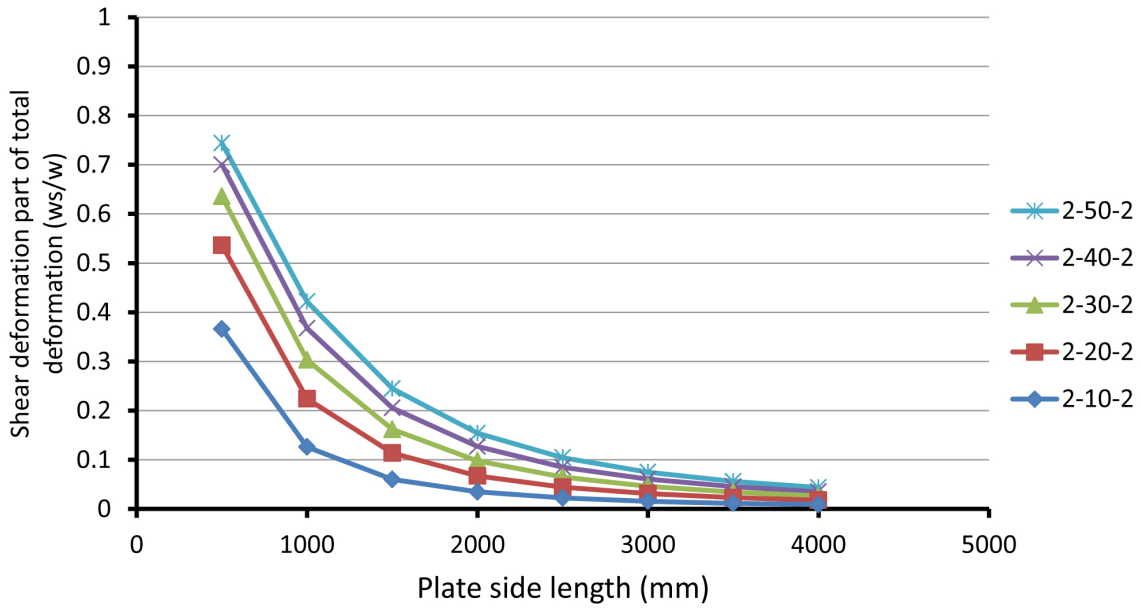


Figure 7.6: Shear deformations as part of total deformation for plates with constant face thickness of 2mm and varying core thickness between 10mm and 50mm.

Just like the effects of varying the faces while keeping the core thickness constant, described in the section above, varying the core thickness will yield similar results. Looking at Fig. 7.6 for the plate with constant face sheets at 2mm, one can see that the shear deformation part of the total deformations will decrease both with increasing in-plane dimensions and decreasing core thickness. The same can be noticed for the plate with thicker face sheets at 5mm, Fig. 7.7, but in this case the shear deformation are for all different scenarios a larger part of the total deformation.

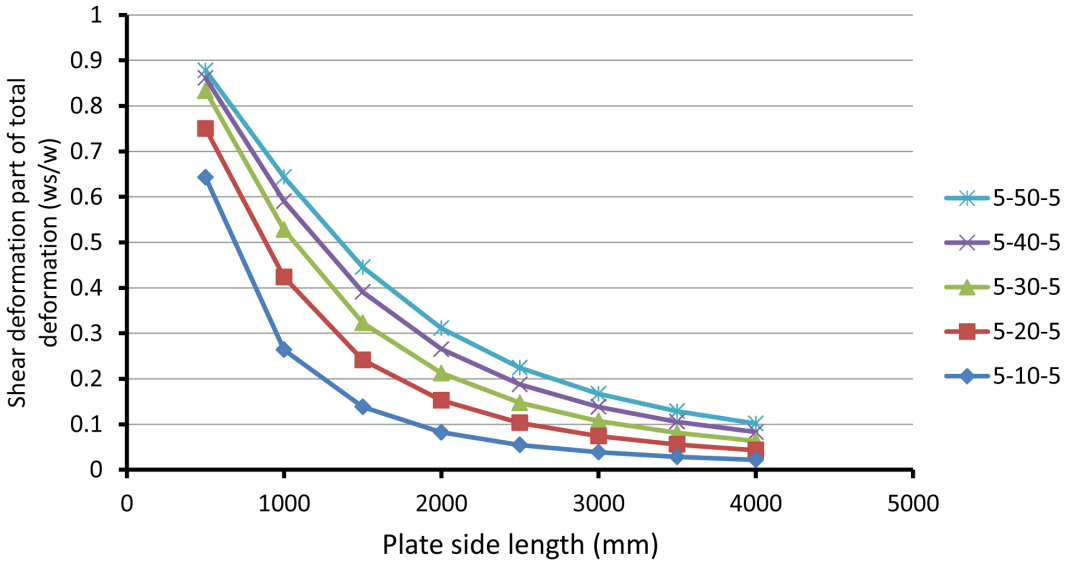


Figure 7.7: Shear deformations as part of total deformation for plates with constant face thickness of 5mm and varying core thickness between 10mm and 50mm.

7.3 Eigenvalue analysis of simply supported plates

7.3.1 Introduction - importance of transverse shear deformations

The relatively weak material of the core will have an impact on the computations of buckling loads, just as it had on the computations of lateral deflections. The lateral deflections will be under-predicted, but the elastic buckling loads will be over-predicted if Kirchhoff thin plate theory is used. The reason for both are the same, namely that failure to include the effects of transverse shear in the core will make a plate too stiff in an analysis. Just as for the lateral deflections, this effect will be prominent for some plates, but not significant for others.

The effect of shear deformations in buckling analyses is shown in Figs. 7.8 and 7.9, where the interaction curve for in-plane normal loads have been plotted for shear deformation theory and thin plate theory, and verified with ABAQUS solid models. The reason for the discrepancy between the analytical and the Finite Element models is that the solid models are a bit softer than the analytical models. This is probably due to the difference in boundary conditions between a simply supported analytical plate model and a solid element model with several layers. A more fitting comparison would be to use shell element models, but then it would

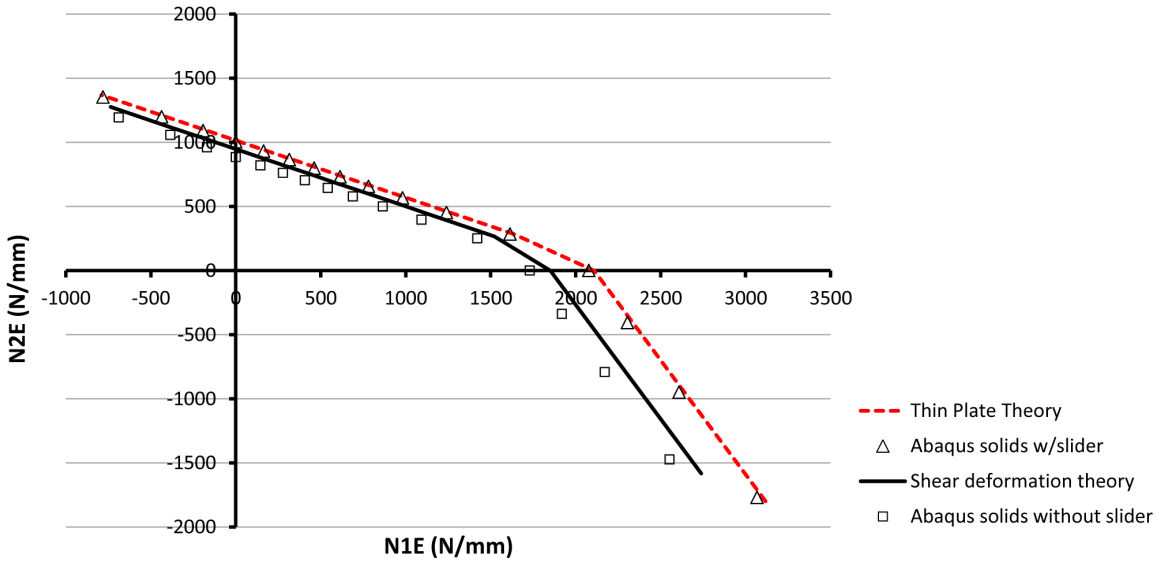


Figure 7.8: Interaction curve for a thin plate at 4190x2800x4-25-4mm with plotted with shear deformation theory presented in this thesis and Kirchhoff thin plate theory. Verification of analytical formula against ABAQUS Finite Element model.

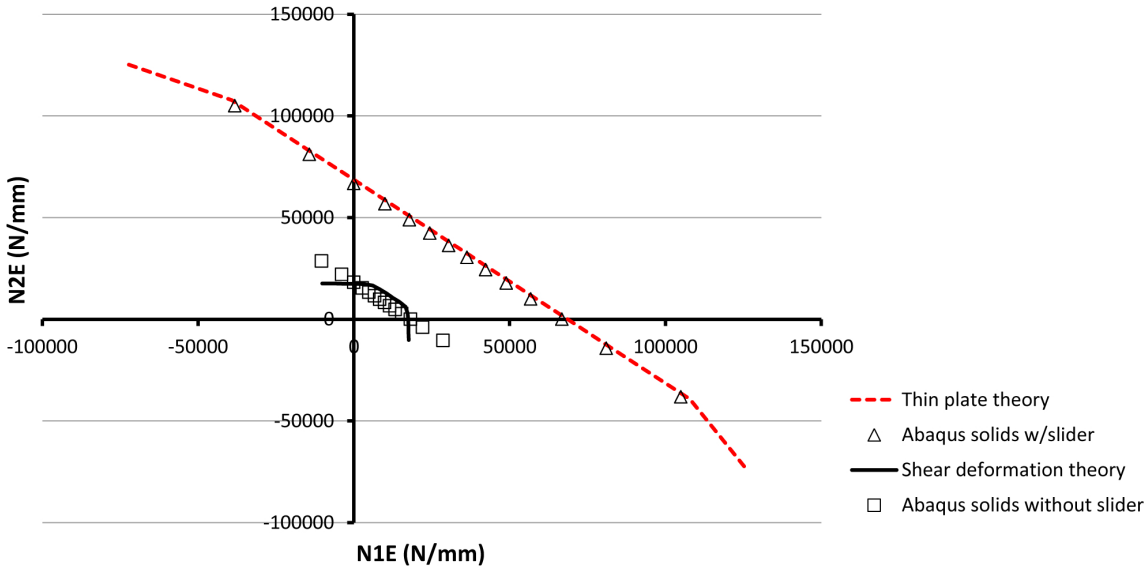


Figure 7.9: Interaction curve for a thin plate at 100x100x5-50-5mm with plotted with shear deformation theory presented in this thesis and Kirchhoff thin plate theory. Verification of analytical formula against ABAQUS Finite Element model.

not be possible to enforce Navier's hypothesis with sliders in ABAQUS, a procedure which was explained in Chapter 5, and the effect of the difference in the two theories would not have been possible to show. It is clear that for this relatively thin plate at 4190x2800x4-25-4mm the difference in using thin plate theory is not that large. The eigenvalues, and hence the buckling loads will be over-predicted, but not significantly. In this case a large enough safety factor in design rules would probably be enough to use thin plate theory safely.

As was found during the pressure analyses in the previous section, however, the transverse shear deformations become increasingly important as the plate gets stockier. In Fig. 7.9 a comparatively thicker plate at 1000x1000x5-50-5mm has been computed using shear deformation and Kirchhoff thin plate theories and the results have been verified against ABAQUS solid element models. It is easy to see the large difference in the interaction curves, using thin plate theory for this kind of stocky plate will significantly over-predict the eigenvalues and hence the buckling loads of the plate. In fact, the eigenvalues computed with thin plate theory are around three times as large as those computed with shear deformation theory. In reality, however, such a thick plate will fail due to material yield at loads far below the elastic buckling loads. It is nonetheless a theoretical result and shows the importance of using this theory when designing SPS sandwich plates.

7.3.2 Verification of analytical model vs. ABAQUS shell elements

The formulae in the analytical model which were presented in Chapter 2 have been checked against ABAQUS shell element models. The plate dimensions are 4190x2800mm and 4-25-4mm thickness. The materials parameters were the same as those used in the convergence analysis, shown in Table 5.1. The plate was loaded in two different biaxial loading scenarios. One with a variation of N1 and N2, i.e. in-plane normal forces, the other with N1 and N3, i.e. one in-plane normal and one in-plane shear force. In case of the first scenario, the applied loads were calculated using points on a circle, starting from -30 degrees all the way up to +120 degrees, with intervals at 10 degrees. In the second case of the second scenario, the initial angle was -120 degrees going all the way up to +120 degrees with 10 degree intervals. In both cases there were thus sufficient points to draw up an accurate enough interaction curve.

The results for the first load scenario are shown in Fig. 7.10. In the figure, the axes are those of the in-plane loads N1 and N2. In the first quadrant of the graph, both forces are in compression, while in the second and fourth quadrants the applied

N1 and N2 forces, respectively, are in tension. Pure uniaxial compression is shown in the intersection points between the curve and the axes. As can be seen in the second and fourth quadrant, tensile forces will stiffen the plate and make it carry larger compressive forces in the perpendicular direction. The change of slope close to the x-axis stems from the fact that this was a non-square sandwich plate. In biaxial loading up to +10 degrees the buckling shape was two half-waves in the longitudinal direction and for load combinations above +10 degrees the buckling shape of the plate was one half-wave in each direction. As one can see there is very good agreement between the analytical formulae and the Finite Element solution for this load scenario.

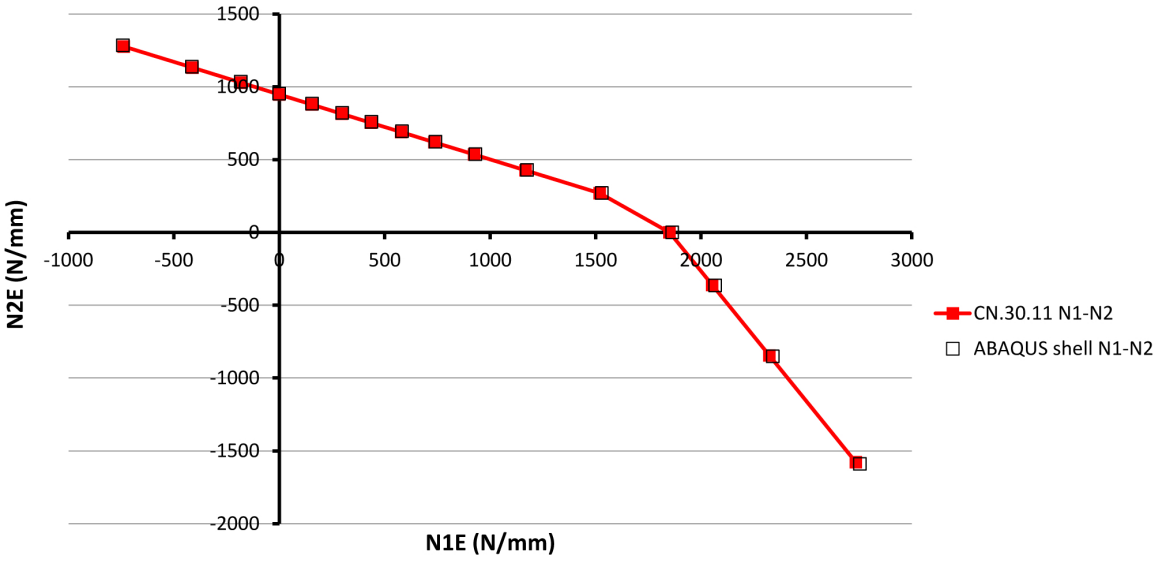


Figure 7.10: Interaction curve for a simply supported 4190x2800x4-25-4mm plate with varying in-plane normal forces N1 and N2. Verification of analytical formula against ABAQUS Finite Element model.

The second load scenario with combination of in-plane normal force and shear force is shown in Fig. 7.11. The intensity of the in-plane normal force N1 is shown on the x-axis, while the intensity of the in-plane shear force N3 is shown along the y-axis. A couple of things are different to the chart from the first load scenario, shown in Fig. 7.10. Firstly, this curve is symmetrical about the x-axis. This stems from the known fact that $\tau_{xy} = \tau_{yx}$, meaning that the direction of the applied shear force is indifferent. The second difference is that this curve is smooth along its entire length. This stems from the fact that a plate buckling under in-plane shear forces cannot be described by the sine functions that are so useful to describe the

buckling shape of a plate loaded by normal forces. This plate buckles into diagonal buckles, see Fig. 7.14a, and there is no marked change from one buckling mode to another. Because the formulae used in computing the buckling factor for a plate in in-plane shear loading (see Section 2.7.3) are only approximate, there is a larger discrepancy between the analytical and the Finite Element solutions, as shown by the markers in Fig. 7.11, than for the first loading scenario. Nonetheless, the agreement is very good, also for this kind of loading.

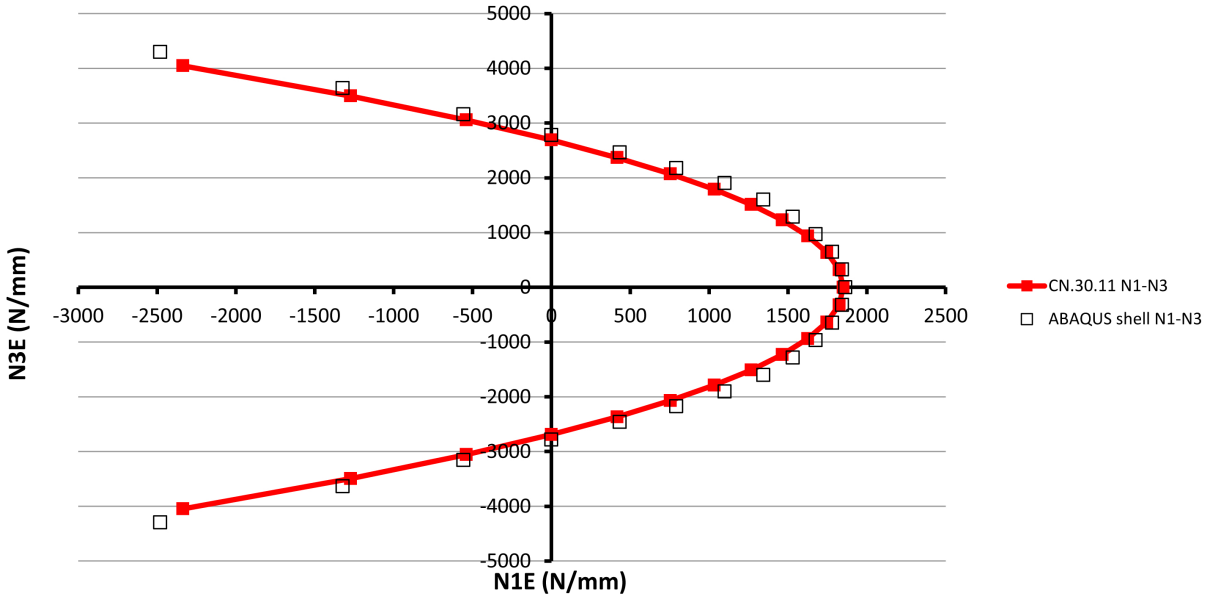
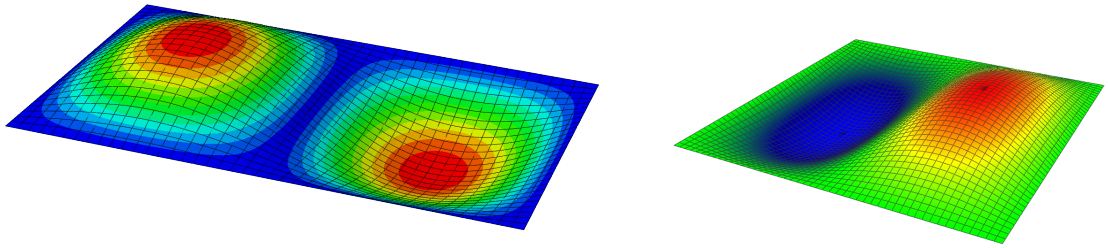


Figure 7.11: Interaction curve for a simply supported 4190x2800x4-25-4mm plate with varying in-plane normal and shear forces N1 and N3. Verification of analytical formula against ABAQUS Finite Element model.

7.3.3 Verification of Rayleigh-Ritz model vs. ABAQUS shell elements

The same plate (4190x2800x4-25-4mm) that was compared to ABAQUS shell models in the previous section has also been used to verify the Rayleigh-Ritz semi-analytical model. The Rayleigh-Ritz model was programmed in FORTRAN. The loading scenarios were the same as for the verification of the analytical formulae in the last section, but with an important addition. Because Rayleigh-Ritz is an energy method, it is easy to include pre-stresses in the model (which was not the case for the analytical models). What this means is that the plate is loaded by one of the three main loads (N1, N2 or N3) before a linear buckling analysis is done. Because the plate is already loaded when this analysis starts, it means that it has



(a) **(b)**

Figure 7.12: (a) Buckling shape plot from ABAQUS for a simply supported sandwich plate in longitudinal buckling and (b) from the Rayleigh-Ritz model in FORTRAN.

already used some of its capacity to carry loads and consequently the eigenvalues should be lower.

The values of the pre-stress (measured in N/mm) were chosen as to correspond with the face sheet thickness. Because the plate considered here had face sheet thickness of 2x4mm, the applied pre-stress were chosen so that by dividing by 8 the stress in the steel plates were, for instance, 1343[N/mm]/8[mm]=166[MPa].

In ABAQUS the pre-stress analyses were done by using a static step with the pre-stress load before the linear perturbation step (buckle analysis), while in Rayleigh-Ritz it was done by inserting a pre-stress matrix into the eigenvalue equations, as shown in Eq. (4.5.1). As one can see from Fig. 7.13 there is very good agreement between the semi-analytical and the Finite Element solutions for the scenario with varying in-plane normal loading, N1 and N2. The other scenario with varying in-plane normal and shear forces, N1 and N3, also shows a very good agreement between the semi-analytical model and the Finite Element solution. It is also clear that for load scenarios that include in-plane shear loading (N3), the Rayleigh-Ritz model will be superior to the approximate formulae in the analytical model. These formulae, however, have the advantage of being very quick to use. It also worth noting that the difference in the two methods is not very large, and that the approximate formulae will yield a conservative eigenvalue.

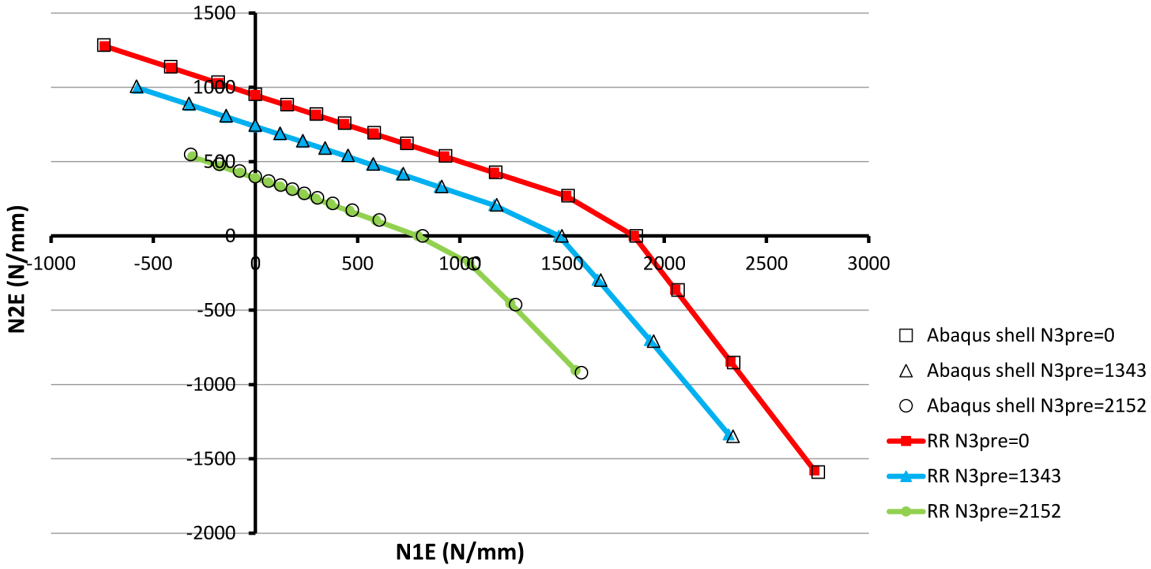
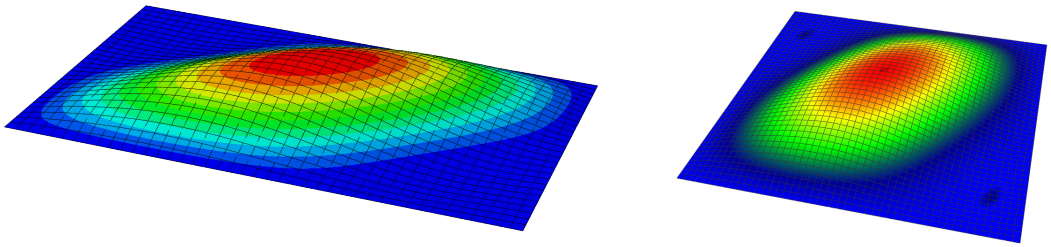


Figure 7.13: Interaction curve for a simply supported 4190x2800x4-25-4mm plate with varying in-plane normal forces N1 and N2. Verification of Rayleigh-Ritz model against ABAQUS Finite Element model.



(a) **(b)**

Figure 7.14: (a) Buckling shape plot from ABAQUS for a simply supported sandwich plate in shear buckling (b) and from the Rayleigh-Ritz model in FORTRAN.

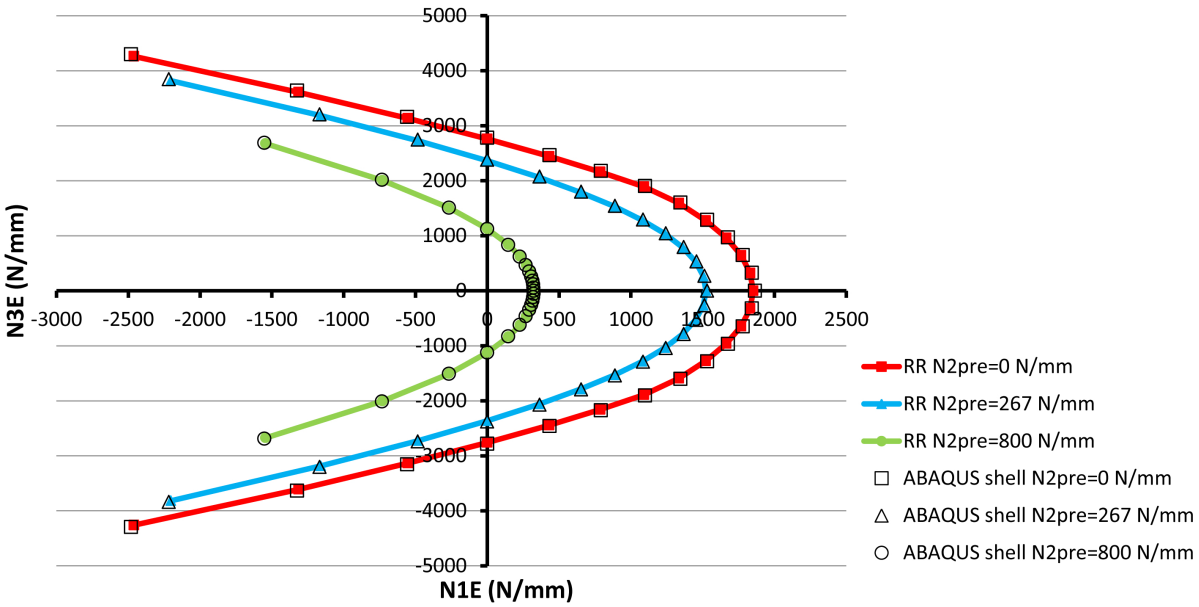


Figure 7.15: Interaction curve for a simply supported 4190x2800x4-25-4mm plate with varying in-plane normal forces N1 and N2. Verification of Rayleigh-Ritz model against ABAQUS Finite Element model.

7.4 Eigenvalue analysis of clamped plates - Rayleigh-Ritz model vs. ABAQUS shell elements

In order to model clamped plates in the Rayleigh-Ritz model, elastic springs were used along the edges of the plate with a very high spring stiffness that made the rotations of the edges approach zero. The magnitude of the spring stiffness was found by simple trial and error in a convergence test; when the eigenvalue had reached a maximum, the springs were stiff enough to prevent edge rotations. The mathematical implementation of these springs is shown in the chapter describing the Rayleigh-Ritz model. For the clamped plates, the eigenvalues did not converge as quickly, i.e. with the same number of terms in the series, as the simply supported plates, and for that reason the computational time of these plates was several times higher. Nonetheless, the total time was still lower than that of analysis in ABAQUS.

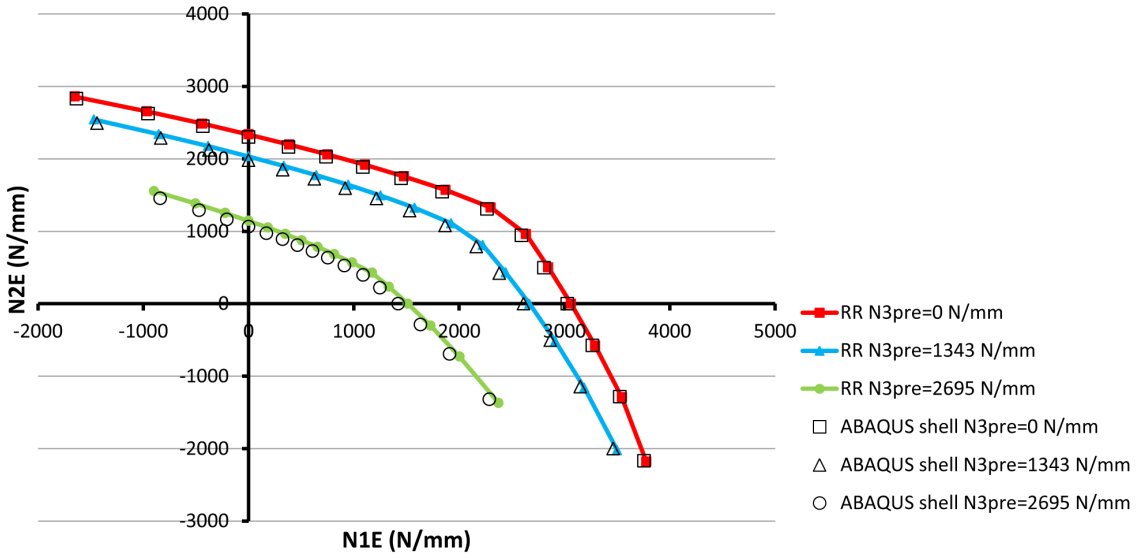
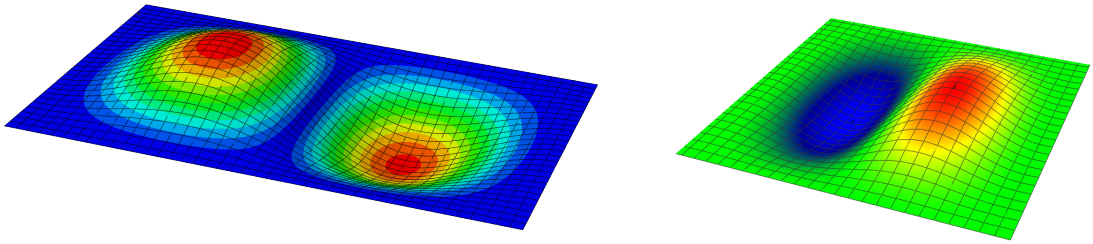


Figure 7.16: Interaction curve for a clamped 4190x2800x4-25-4mm plate with varying in-plane normal forces N1 and N2. Verification of Rayleigh-Ritz model against ABAQUS Finite Element model.

The same plate as was used for the simply supported plate verifications (4190x2800x4-25-4mm) was used here as well. Clamping the plate with springs will make it stiffer and able to carry higher in-plane loads. The load scenarios were the same as for the simply supported plates, one with varying in-plane normal forces, N1 and N2, and one with varying in-plane normal and shear forces, N1 and N3. Pre-stressing



(a) **(b)**

Figure 7.17: (a) Buckling shape plot from ABAQUS for a clamped sandwich plate in longitudinal compression and (b) from the Rayleigh-Ritz model in FORTRAN.

of the plate was also done for the clamped models.

The results for the first load scenario with varying in-plane normal forces, N1 and N2, are shown in Fig. 7.16 and the comparison of the buckling shape from ABAQUS and Rayleigh-Ritz is shown in 7.17a. The curve is smoother compared with the simply supported plate with same loading in Fig. 7.13. This is due to the fact that for a simply supported plate, the region in the load-space where the buckling shape is similar, the curves are linear. For the clamped plate, the plate buckling shape is not described by sine half-waves, and it may be that for the clamped plate the change between different modes is not as pronounced as for the simply supported plate. Again, there is very good agreement between the semi-analytical and Finite Element solutions.

For the second load scenario with varying in-plane normal force, N1, and in-plane shear force, N3, the results are shown in 7.18 and the corresponding buckling shapes from ABAQUS and Rayleigh-Ritz are shown in Figs. 7.19a and 7.19b. As for the first scenario, there is very good agreement between the Finite Element method and the Rayleigh-Ritz method.

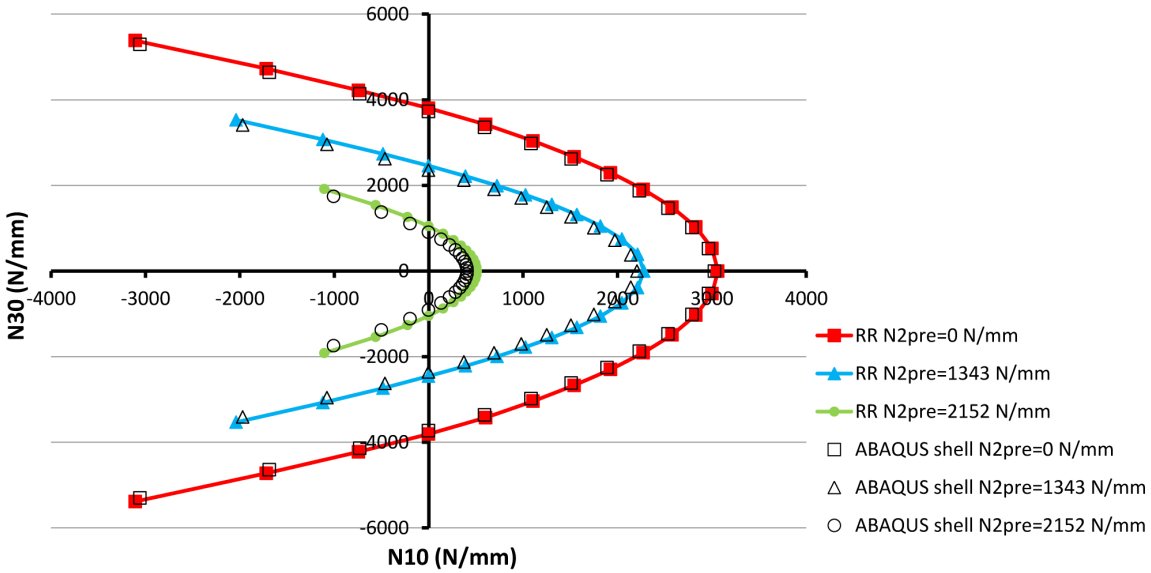
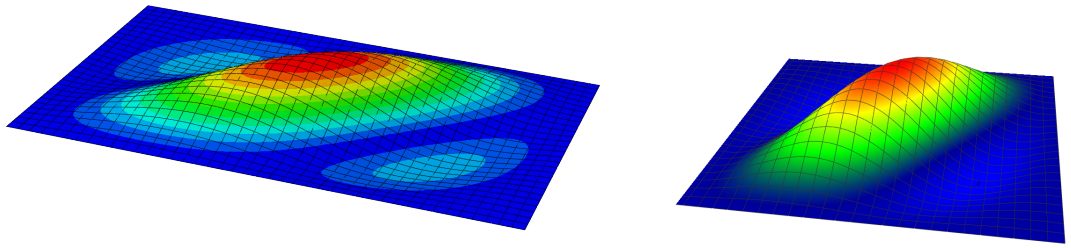


Figure 7.18: Interaction curve for a clamped 4190x2800x4-25-4mm plate with varying in-plane normal forces N1 and N3. Verification of Rayleigh-Ritz model against ABAQUS Finite Element model.



(a) **(b)**

Figure 7.19: (a) Buckling shape plot from ABAQUS for a clamped sandwich plate in shear buckling and (b) from the Rayleigh-Ritz model in FORTRAN.

7.5 Elasto-plastic buckling analysis

7.5.1 Elasto-plastic formula vs. eigenvalue and von Mises

The elasto-plastic buckling formula in Eq. (2.8.6) is a simplified ultimate capacity check. It corrects the elastic eigenvalue for material yield and initial imperfection. It does not take the reserve capacity of plates, or the strain hardening of the steel, into account. For the 4180x2800x4-25-4mm plate, the formula has been plotted in Fig. 7.20 against the eigenvalue computed with the analytical formula, Eq. (2.7.11) and the von Mises ellipse. The von Mises ellipse indicates the points on the load curve where the plate will reach its ultimate capacity due to material yield of the face plate material. The fit between the elasto-plastic formula and the eigenvalue and von Mises curves is generally good with the elasto-plastic buckling curve always being on the inside, or on top of, the two other curves.

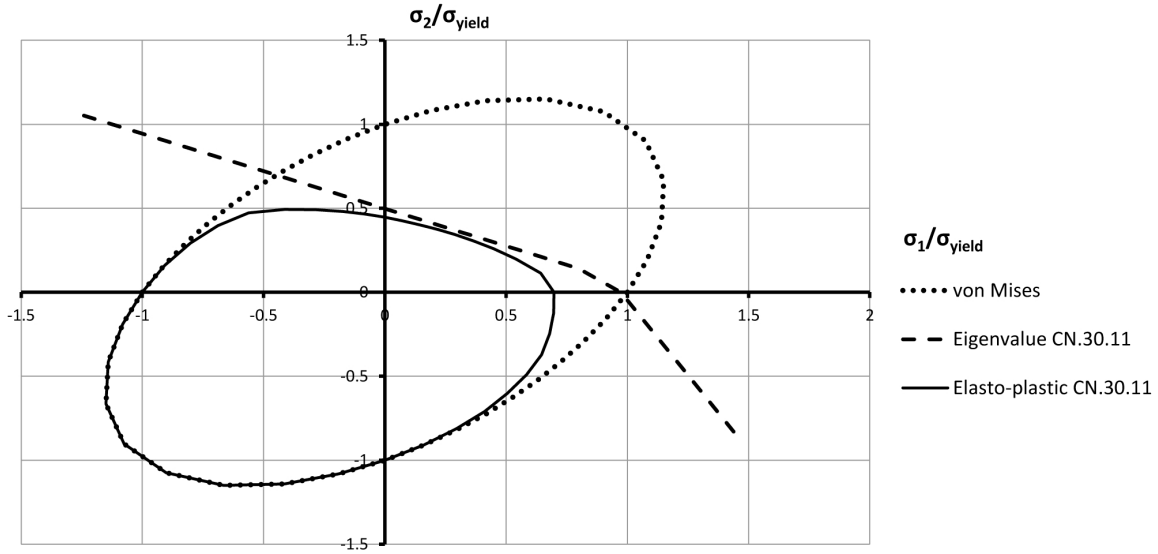


Figure 7.20: Capacity curve with elasto-plastic loads compared with von Mises yield and eigenvalues, all computed with the analytical formulae. All values have been normalised with respect to the yield stress in the face plate material.

7.5.2 Elasto-plastic formula vs. ABAQUS ultimate capacity

The elasto-plastic formula has been compared with a solid element model in ABAQUS. In this model the face plate steel is a bilinear material with a strain-hardening parameter $E_T = 1000MPa$. This model was run through a series of non-linear Riks analyses in ABAQUS, thus taking into account both the material plasticity and the reserve capacity in the plate. For the non-linear analyses the imperfections used were the first buckling modes from the linear buckling analysis. The script used to make the input files for ABAQUS is shown in Appendix C.2.

The shape and the amplitude of the imperfection can greatly influence the results in a non-linear analysis. In Fig. 7.21 the ultimate capacity from the ABAQUS solid model has been plotted for two different imperfection scenarios along with the same curves that are shown in Fig. 7.20. The two ultimate capacity curves represent two different analysis methods. One where the imperfection from the first buckling analysis, at -30 degrees on the load interaction curve, was used for all the subsequent analyses, thus making it necessary to run only one linear buckling analysis, and one where a new buckling analysis was run before each non-linear analysis, thus obtaining the correct buckling shape as a basis for the imperfection of each non-linear case. This was done in order to investigate the jump in the blue curve (the one with a new imperfection for each case) around the x-axis of Fig. 7.21. This area of the curves is where the buckling shape changes from two half-waves to one half-wave in the longitudinal direction, and it was thought that this jump might not indicate the smallest ultimate capacity.

The plotting of the beige line in Fig. 7.21 confirmed that this was indeed the case. This line represents the ultimate capacity computed with the same imperfection shape for all analyses. It is seen that for the load cases below the x-axis the blue and beige lines are indeed the same, because at these load cases the number of half-waves are similar for both scenarios and the imperfection shapes are identical. As the load approaches +30 degrees where the two curves intersect, it is clearly the blue line (updated imperfection) that yields the lowest ultimate capacity for the next load cases. The reason for the beige line's over-prediction is that because the plate is forced to deform into a shape with two half-waves it is made unnaturally stiff at these load combinations, essentially requiring that the transverse deflection, w, is zero along the mid-span of the L1-direction. As the curves approach +180 degrees, they seem to converge, but as the applied stresses are increasingly tensile, this particular model becomes increasingly invalid. In this thesis, however, the interesting part of the curve is the one plotted, between -30 and +120 degrees on the load interaction curve. In this part of the load space, the

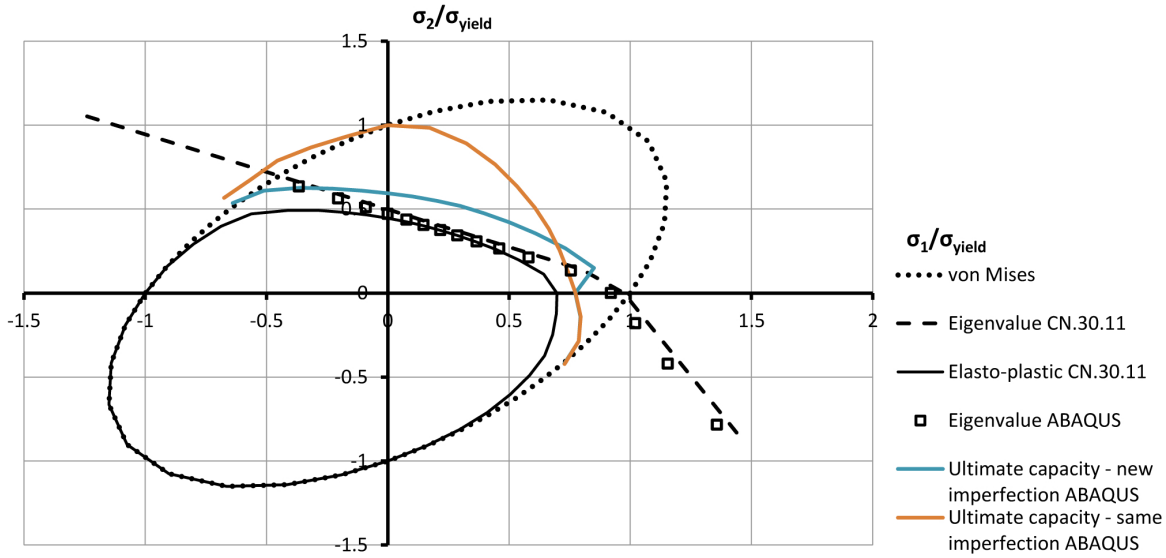


Figure 7.21: Interaction curve with von Mises stress, elasto-plastic and ultimate capacity and eigenvalues. All values have been normalised with respect to the yield stress in the face plate material.

ultimate capacity is the minimum of the two curves, which, except for the small jump around 0 to +10 degrees, coincides with the blue curve.

It is also interesting to compare the ultimate capacity and eigenvalue curves from ABAQUS. In uniaxial compression in the longitudinal direction, the elastic buckling load is higher than the ultimate capacity, but for the other load cases, above +10 degrees on the load interaction curve, the ultimate capacity of the plate is the highest. The reason is that because the plate is non-square, the reduced slenderness of the plate is different in the two principal load directions. Jumping ahead and looking at Fig. 7.23 one can see that for small values (1.5 and lower) of the reduced slenderness $\bar{\lambda}$ the eigenvalue might be much higher than the elasto-plastic value, or they might be fairly close, as they are from the region around 1.5. At values below 1, the plate capacity is dominated by material yield, while at values higher than 1 the elastic eigenvalue is the dominant factor. In the eigenvalue dominated region, where the reduced slenderness is below 1, non-linear analyses will also include the reserve capacity of plates, something which is evident when comparing the ABAQUS ultimate capacity and the elasto-plastic values in figure 7.23. These effects are not accounted for by the elasto-plastic formulae used in this thesis and in the DNV classification note on Sandwich Panels [1].

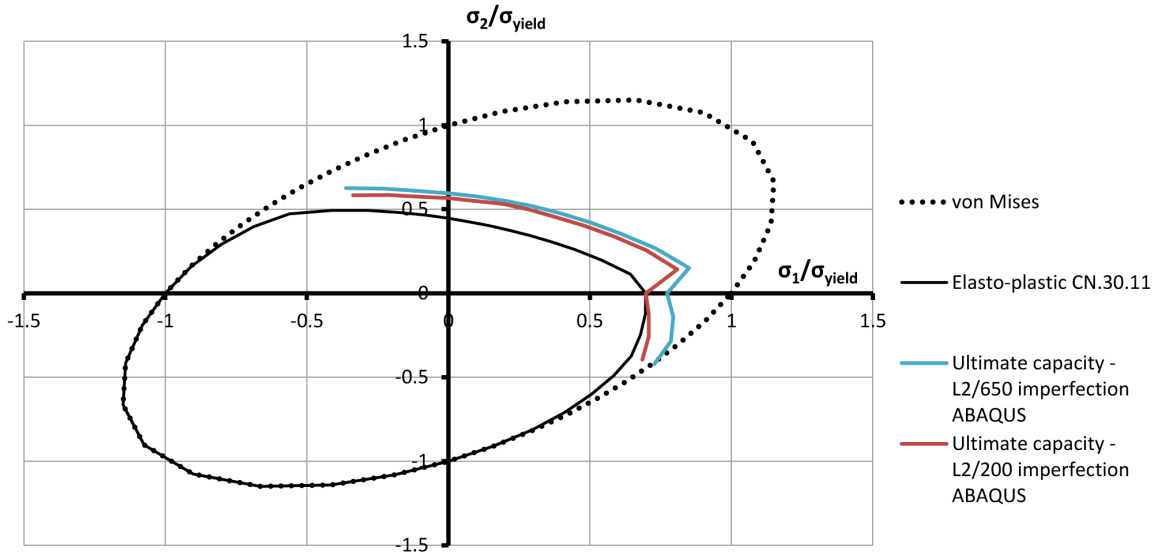


Figure 7.22: Interaction curve with the von Mises ellipse, the elasto-plastic buckling load and two different ultimate capacity curves from ABAQUS which are dependent on the amplitude of the imperfection in the analyses. Two different imperfection amplitudes are used, $L2/200mm$ and $L2/650mm$.

The amplitude of the imperfection has also been considered. In Fig. 7.22 the blue ultimate capacity curve in Fig. 7.21 has been plotted for an imperfection amplitude of $L2/200$ and $L2/650$, where $L2$ is the length of the shortest edge of the plate. As can be seen there is a larger difference where the imperfection shape is made up of two half-waves, than where there is only one half-wave.

7.5.3 Reduced slenderness curve

The elasto-plastic formula has also been plotted against the analytical eigenvalue and the ABAQUS eigenvalue and ultimate capacity values in Fig. 7.23. The same in-plane dimensions were used for these analyses as for the other buckling and ultimate strength analyses as above, but the core was varied in thickness between 80mm and 10mm with steps of 5mm. The curve is seen to give a very good approximation of the ultimate capacity for most cases. As the reduced slenderness, $\bar{\lambda}$, increases beyond 1 it can be seen that the elasto-plastic formula yields lower values than the ultimate capacity of ABAQUS. The reason is that this formula does not allow loads beyond elastic buckling and hence the reserve capacity which is found in plates is not accounted for. As the reduced slenderness approaches 0.5

it can be seen that the elasto-plastic curve converges towards the yield strength of the material while the ABAQUS results suddenly rise above this line. This is the effect of strain hardening in the steel. These effects have not been investigated further in this thesis, but the point of the elasto-plastic formula is to be a safe and easy-to-use approximation and, because it does not take any of these effects into consideration, it is.

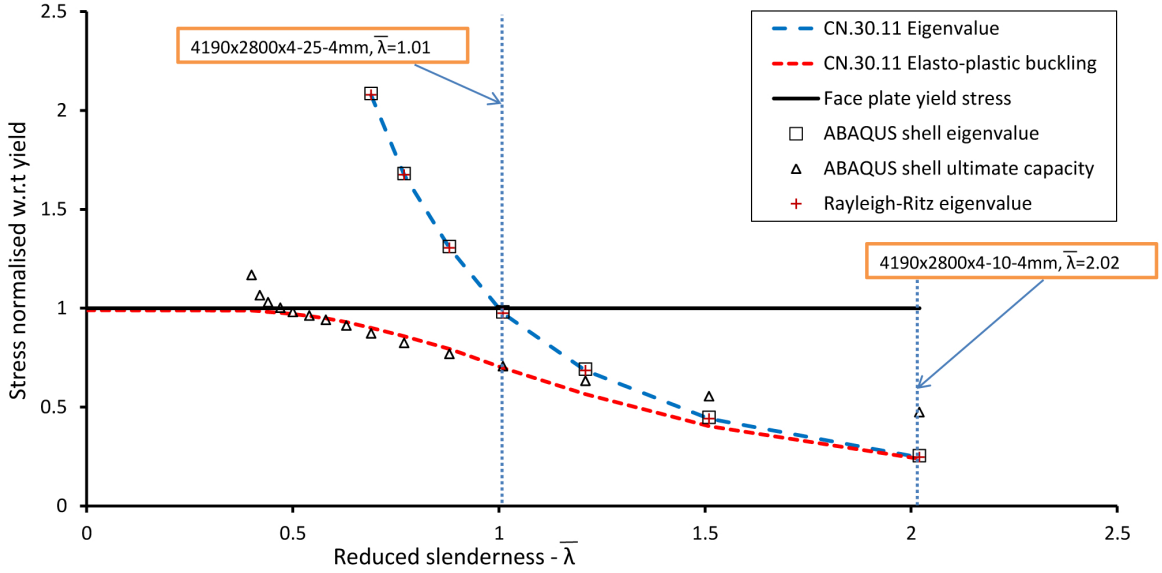


Figure 7.23: Capacity versus reduced slenderness plot of elastic buckling loads, elasto-plastic load and ultimate capacity. Plate is subjected to uniaxial compression in the longitudinal direction.

7.5.4 Load-displacement curves

ABAQUS has been used to create non-linear load-displacement curves for different plates. Two of these plates are included here. The results are the same as used for plotting the reduced slenderness curve in Fig. 7.23. Whereas, in that curve only the ultimate load was extracted, here the data are used to plot a load-displacement path.

Fig. 7.24 shows one such graph plotted for three different levels of imperfection amplitudes, $L2/2000$, $L2/500$ and $L2/200$. Along with these three curves, the elastic buckling load and the elasto-plastic buckling loads have been plotted. This is a plate with a reduced slenderness of around 2.0 so according to Fig. 7.23 there should be a large reserve capacity above the eigenvalue, and also the elasto-plastic

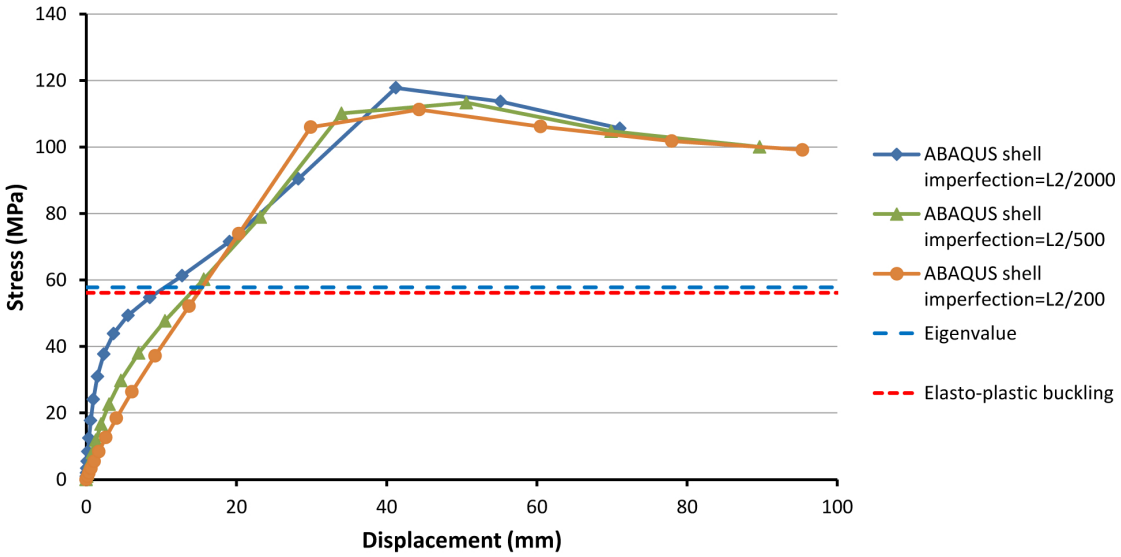


Figure 7.24: Load-displacement curve for a plate with reduced slenderness of about 2.

buckling load.

Fig. 7.25 shows the same load-displacement curve for a stockier plate, with a reduced slenderness of around 1.0. The same three imperfection amplitudes have been plotted. Here the eigenvalue is higher than the ultimate capacity found in ABAQUS, while the elasto-plastic buckling value touches the ultimate capacity of the largest imperfection amplitude at $L2/200$.

What is also apparent is that for the slender plate, there is little difference between the three imperfection curves, whereas for the stockier plate there is difference both in shape and ultimate capacity. This indicates that the stockier plate is more sensitive to the amplitude of the imperfections, but in reality this will be countered by the fact that the geometrical imperfections are smaller and less

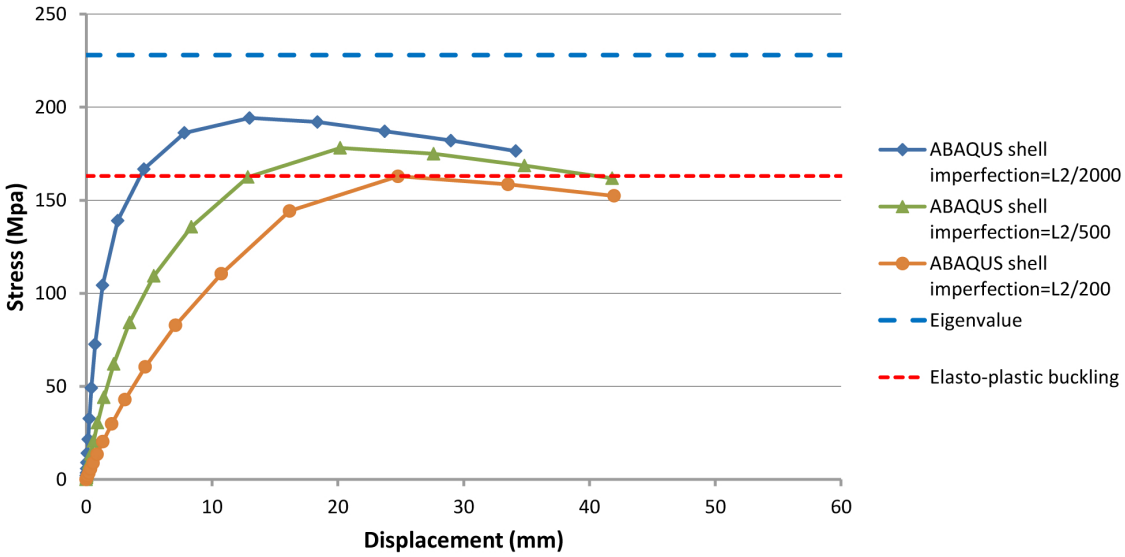


Figure 7.25: Load-displacement curve for a plate with reduced slenderness of about 1.

Chapter 8

Conclusions

8.1 Introduction

In this thesis alternative methods to the Finite Element method have been found and verified for bending and buckling of steel elastomer sandwich plates. Because the material of the core layer of this type of sandwich plate is so weak compared to the face plate material, significant transverse shear deformations can arise in the core. For this reason the normal Kirchhoff thin plate theory can not be relied upon to yield satisfactory results in plate problems such as bending deflection and eigenvalues for buckling.

Two methods have been considered, Mindlin-Reissner plate theory, which is an analytical method that extends the Kirchhoff thin plate theory to include the effects of transverse shear deformations, and a semi-analytical Rayleigh-Ritz model. The analytical formulae have been implemented in a automated Excel sandwich plate calculator and the Rayleigh-Ritz model has been implemented into FORTRAN as a code. The Rayleigh-Ritz model uses Fourier series to approximate the problem variables. The analytical model can be used to check for lateral deflections and buckling loads in simply supported plates. The Rayleigh-Ritz model does not check for lateral deflections due to transverse pressure, it is a pure eigenvalue solver. On the other hand it can model both simply supported and clamped plates via elastic springs at the boundary edges, and indeed any elastic stiffness between the simply supported and clamped cases. Prestress is also included in the Rayleigh-Ritz model, as is a more accurate approximation of eigenvalues where in-plane shear loading is present, than the analytical closed form model.

In addition to this, a conservative ultimate capacity formula has been checked that takes the elastic eigenvalues found in the analytical models and corrects these

for material yield. All the results from the different models have been verified against the Finite Element software ABAQUS.

The results generally show very good agreement, with some small difference for some cases. As an alternative to the Finite Element method, both the analytical formulae implemented in Excel, and the semi-analytical model implemented in FORTRAN will yield satisfactory results for the problems discussed in this thesis, and often at a considerable lower cost of modelling and computational time.

8.2 Analytical models

The Mindlin-Reissner plate theory is a theory originally conceived for thick isotropic plates, but by using stiffness coefficients computed for a sandwich plate, this method is applicable also for the type of sandwich plates considered in this thesis. In Chapter 2 the differential equation for a sandwich plate was derived using summation of forces and moments, and this equation was solved in order to arrive at expressions for bending deflection and eigenvalues. The principle of partial deflections was also introduced, which states that the plate problems can be solved for bending and shear deformations separately and the results superimposed to get the total deformations. The expressions for eigenvalue are exact for applied in-plane normal forces, but for applied in-plane shear forces an empirical formula must be used. Formulae for combined loading in the buckling analyses are also introduced. Much of the theory in Chapter 2 is shared with the new DNV classification note on sandwich plates [1], and has been implemented in an automated Excel sandwich plate calculator, as shown in Chapter 6.

The results of the comparison between the analytical closed form solutions and the Finite Element method are shown in Chapter 7. For the lateral pressure analyses several plates were considered with varying ratio between core and face sheet thickness, as well as varying face sheet thickness for a constant core thickness and vice versa. In these analyses ABAQUS solid models were used in order to find the total deformation, the bending deformation and shear deformation, and these were compared to the analytical values.

For the core and face ratio analyses, the factor that most influenced the deformations was the reduction of the face sheet thickness. The bending deformation part increased between 50-100% for the considered plates, while the shear deformation part was virtually unchanged.

In the analyses were the core thickness was kept constant and the face sheets

were varied, and vice versa, the shear deformation was found to often be a large part of the total deformation. This depended most heavily on the slenderness of the plate. In all cases there was a good agreement between the analytical model and the ABAQUS solid model, the small difference might be attributed to the differences in boundary conditions between a analytical plate and a solid element plate with several layers.

For the buckling analyses the analytical model yielded very good agreement with to ABAQUS for the case where only in-plane normal forces were applied. The case where in-plane shear forces are also present had some disagreement between the two methods, but it must be stated again that for shear buckling, an empirical formula had to be used in the analytical model.

The analytical model was finally used in a simplified ultimate capacity check, where the elastic eigenvalues were corrected for material yield in the face plate. In this check, the ultimate capacity can never exceed either elastic eigenvalues or von Mises yield. Compared to ABAQUS ultimate capacity analyses done with geometric and material non-linearity, this was shown as a safe and accurate approach for plates with reduced slenderness below 1, while for thinner plates, the method heavily under-predicted the ultimate capacity compared to ABAQUS. The reason is that effects such as strain hardening or plate reserve capacity is not considered in the simplified method.

8.3 Semi-analytical model - Rayleigh-Ritz

The principles of the Rayleigh-Ritz method are shown in Chapter 4 and the particular model developed for sandwich plates is shown in Chapter 4. The method springs out from the potential energy expression of the plate. By using the Fourier series to approximate the transverse deflection and shear deformations, this method will transform the continuous variables in the differential equation into a set of algebraic equations. The method is very powerful as is evident by the relative ease with which both elastic springs at the boundary and prestresses have been included. This method is also much better than the analytical formulae when applied in-plane shear forces are present in buckling analyses. Because both the Rayleigh-Ritz and Finite Element methods springs out of the same principle, or theory, this was somewhat expected.

The results from the verification of the method are shown in Chapter 7. The model was used to check buckling from in-plane normal and shear forces, and the results were verified against ABAQUS shell element models. For the simply

supported plates, the agreement between the two methods was very good, both in cases with prestress and where in-plane shear forces were present. Especially in the latter, the Rayleigh-Ritz was more accurate than the analytical formulae. For the clamped plates, the agreement was also good, but the computational time in FORTRAN was several times higher than for the simply supported plates. Nonetheless, even the clamped plates were faster in FORTRAN than in ABAQUS because the method does not require time for modelling of each case.

8.4 Suggestions for further work

During the work of this thesis, several ideas for further work was discussed. They are mainly related to the Rayleigh-Ritz model, as this is the one with the most potential. Examples of further work include:

- Extend the Rayleigh-Ritz model to also include stiffeners in the plate. Steel elastomer sandwich plates are ideal for use in repair of damaged structures, and this will most often include some kind of stiffened plate. Example of Rayleigh-Ritz models for stiffened plates are given in Brubak (2003) [18] and Brubak, Hellesland and Steen [18].

- Extend the Rayleigh-Ritz model to a non-linear model which can be used in ultimate capacity analyses, as was done in Brubak [19].

- Connect the Rayleigh-Ritz model to the Excel spread-sheet calculator, so that the buckling part of the calculator is done with semi-analytical methods. This can be advantageous for cases with in-plane shear forces, or for prestress.

- Creating a sandwich/thick plate element for PULS based on the Rayleigh-Ritz model in this thesis has also been discussed as a suggested future work.

References

[1] Det Norske Veritas AS. *Classification Note - No. 30.11 - Steel Sandwich Panel Construction*. DNV AS, Høvik, Norway, 2012.

[2] Jostein Fladby. *Ultimate Strength of Steel-Elastomer Sandwich Panels Under Combined Loadings Including In-Plane Shear*. MSc Thesis, Mechanics Division, Department of Mathematics, University of Oslo, 2010.

[3] Dan Zenkert. *An Introduction to Sandwich Structures - Student Edition*. Dan Zenkert, 2005.

[4] J. N. Reddy. *Theory and Analysis of Elastic Plates and Shells*. CRC Press, 2007.

[5] Stephen P. Timoshenko and S. Woinowsky-Krieger. *Theory of Plates and Shells*. McGraw-Hill Book Company, Inc., 1959.

[6] Don O. Brush and Bo O. Almroth. *Buckling of Plates, Bars and Shells*. McGraw-Hill Book Company, 1975.

[7] N. J. Hoff. *Bending and Buckling of Rectangular Sandwich Plates*. National Advisory Commitee for Aeronautics, 1950.

[8] Zdeněk P. Bažant and Luigi Cedolin. *Stability of Structures*. World Scientific, 2010.

[9] Edward W. Kuenzi, W. S. Ericksen, and John J. Zahn. *Shear Stability of Flat Panels of Sandwich Construction*. Forest Products Laboratory, United States Department of Agriculture, Madison, Wisconsin, 1962.

[10] C. S. Smith. *Design of Marine Structures in Composite Materials*. Elsevier Science Publishers LTD, Essex, England, 1990.

[11] Lars Brubak. *Knekning av plater of skall i skipskonstruksjoner*. Cand. Scient. thesis, Mechanics Division, Department of Mathematics, University of Oslo, 2003.

[12] Robert D. Cook, David S. Malkus, Michael E. Plesha, and Robert J. Witt. *Concepts and Applications of Finite Element Analysis, Fourth Edition.* John Wiley & Sons, Inc, 2002.

[13] Pål G. Bergan and Tor G. Syvertsen. *Knekning av søyler og rammer.* Tapir forlag, 1989.

[14] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method - Its Basis & Fundamentals 6th Edition.* Elsevier, 2008.

[15] Simulia. *Abaqus 6.10 Analysis User's Manual.* Dassault Systèmes, 2010.

[16] Eric Qiuli Sun. *Shear Locking and Hourglassing in MSC Nastran, ABAQUS and ANSYS.* MSC.Software's Virtual Product Development Conferences, 2006.

[17] Air Force Research Lab Lt. Jeff Brown. *Characterization of MSC/NASTRAN and MSC/ABAQUS Elements for Turbine Engine Blade Frequency Analysis.* 1997 Aerospace Users' Conference Proceedings, 1997.

[18] Jostein Hellesland Lars Brubak and Eivind Steen. *Semi-analytical buckling strength analysis of plates with arbitrary stiffener arrangements.* Journal of Constructional Steel Research, 2007.

[19] Lars Brubak and Jostein Hellesland. *Strength criteria in semi-analytical, large deflection analysis of stiffened plates in local and global bending.* Thin-Walled Structures, 2008.

Appendix A

Rayleigh-Ritz model

A.1 Differentiated series expressions

The derivatives of the assumed displacement functions for transverse deflection w, and the cross-sectional rotations γ_{xz} and γ_{yz} are used in the expressions for the potential energy of the plate. The differentiated expressions are:

$$\begin{aligned}
\frac{\partial w}{\partial x} &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\left(\frac{m\pi}{a}\right) cos\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\
\frac{\partial w}{\partial y} &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\left(\frac{n\pi}{b}\right) sin\frac{m\pi x}{a} cos\frac{n\pi y}{b} \\
\frac{\partial^2 w}{\partial x^2} &= -\sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\left(\frac{m\pi}{a}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\
\frac{\partial^2 w}{\partial y^2} &= -\sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\left(\frac{n\pi}{b}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\
\frac{\partial^2 w}{\partial x\,\partial y} &= \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} A_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right) cos\frac{m\pi x}{a} cos\frac{n\pi y}{b} \\
\frac{\partial \gamma_{xz}}{\partial x} &= -\sum_{m=1}^{\infty}\sum_{n=1}^{\infty} B_{mn}\left(\frac{m\pi}{a}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b} \\
\frac{\partial \gamma_{yz}}{\partial y} &= -\sum_{m=1}^{\infty}\sum_{n=1}^{\infty} C_{mn}\left(\frac{n\pi}{b}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}
\end{aligned} \tag{A.1.1}$$

$$\frac{\partial \gamma_{xz}}{\partial y} = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} B_{mn}\left(\frac{n\pi}{b}\right) cos\frac{m\pi x}{a} cos\frac{n\pi y}{b}$$
$$\frac{\partial \gamma_{yz}}{\partial x} = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty} C_{mn}\left(\frac{m\pi}{a}\right) cos\frac{m\pi x}{a} cos\frac{n\pi y}{b}$$

A.2 Expressions for potential energy in Rayleigh-Ritz model

The Rayleigh-Ritz method rarely yields the exact solution. Rather, the computed eigenvalues converge from above toward the exact eigenvalues as more terms in the displacement series are added. These functions are approximated by adding a finite number of terms in the series, from 1 up to M, N, P and Q, respectively. When the displacement functions have been inserted into the expressions for strain energy and load potential, these become:

$$\begin{aligned}
U = \frac{1}{2}\int_{y=0}^{b}\int_{x=0}^{a}\Bigg[& D_x\left(-\sum_{m=1}^{M}\sum_{n=1}^{N} A_{mn}\left(\frac{m\pi}{a}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right. \\
& \left. + \sum_{m=1}^{M}\sum_{n=1}^{N} B_{mn}\left(\frac{m\pi}{a}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right)^2 \\
& + (D_x\nu_{yx} + D_y\nu_{xy})\left(-\sum_{m=1}^{M}\sum_{n=1}^{N} A_{mn}\left(\frac{m\pi}{a}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right. \\
& \left. + \sum_{m=1}^{M}\sum_{n=1}^{N} B_{mn}\left(\frac{m\pi}{a}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right) \\
& \left(-\sum_{m=1}^{M}\sum_{n=1}^{N} A_{mn}\left(\frac{n\pi}{b}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right. \\
& \left. + \sum_{m=1}^{M}\sum_{n=1}^{N} C_{mn}\left(\frac{n\pi}{b}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right) \\
& + Dy\left(-\sum_{m=1}^{M}\sum_{n=1}^{N} A_{mn}\left(\frac{n\pi}{b}\right)^2 sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right. \\
& \left. + \sum_{m=1}^{M}\sum_{n=1}^{N} C_{mn}\left(\frac{n\pi}{b}\right) sin\frac{m\pi x}{a} sin\frac{n\pi y}{b}\right)^2
\end{aligned}$$

$$+\frac{D_{xy}}{2}\left[\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right.\right.$$
$$\left.-\sum_{m=1}^{M}\sum_{n=1}^{N}C_{mn}\left(\frac{m\pi}{a}\right)cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right)$$
$$+\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right.$$
$$\left.\left.-\sum_{m=1}^{M}\sum_{n=1}^{N}B_{mn}\left(\frac{n\pi}{b}\right)cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right)\right]^2$$
$$+S_x\left(\sum_{m=1}^{M}\sum_{n=1}^{N}B_{mn}\,cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right)^2+S_y\left(\sum_{m=1}^{M}\sum_{n=1}^{N}C_{mn}\,sin\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right)^2\Bigg]dxdy \tag{A.2.1}$$

$$W=\frac{1}{2}\int_{y=0}^{b}\int_{x=0}^{a}\left[-2q\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\,sin\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right.$$
$$+N_x\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{m\pi}{a}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right)^2 \tag{A.2.2}$$
$$+N_y\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{n\pi}{b}\right)sin\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right)^2$$
$$\left.+2N_{xy}\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{m\pi}{a}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}\right)\left(\sum_{m=1}^{M}\sum_{n=1}^{N}A_{mn}\left(\frac{n\pi}{b}\right)sin\frac{m\pi x}{a}cos\frac{n\pi y}{b}\right)\right]dxdy$$

The occurrence of the series in the second power, prompts the using of four rather than two series when the expressions are expanded, summing over m, n, p and q. Using the trigonometric expressions for simplification:

$$[SIN]=sin\frac{m\pi x}{a}sin\frac{n\pi y}{b}sin\frac{p\pi x}{a}sin\frac{q\pi y}{b} \tag{A.2.3}$$
$$[COS]=cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}cos\frac{p\pi x}{a}cos\frac{q\pi y}{b} \tag{A.2.4}$$

$$
\begin{aligned}
U = \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q}\int_{y=0}^{b}\int_{x=0}^{a}\Bigg[\Bigg[\Bigg[D_x\Bigg[A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)^2\left(\frac{p\pi}{a}\right)^2 - A_{mn}B_{pq}\left(\frac{m\pi}{a}\right)^2\left(\frac{p\pi}{a}\right) \\
- A_{pq}B_{mn}\left(\frac{m\pi}{a}\right)\left(\frac{p\pi}{a}\right)^2 + B_{mn}B_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{p\pi}{a}\right)\Bigg] \\
+ (D_x\nu_{yx} + D_y\nu_{xy})\Bigg[A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)^2\left(\frac{q\pi}{b}\right)^2 - A_{mn}C_{pq}\left(\frac{m\pi}{a}\right)^2\left(\frac{q\pi}{b}\right) \\
- B_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{q\pi}{b}\right)^2 + B_{mn}C_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{q\pi}{b}\right)\Bigg] \\
+ D_y\Bigg[A_{mn}A_{pq}\left(\frac{n\pi}{b}\right)^2\left(\frac{q\pi}{b}\right)^2 - A_{mn}C_{pq}\left(\frac{n\pi}{b}\right)^2\left(\frac{q\pi}{b}\right) \\
- A_{pq}C_{mn}\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right)^2 + C_{mn}C_{pq}\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right)\Bigg]\Bigg][SIN] \\
+ \Bigg[\frac{D_{xy}}{2}\Bigg[A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{p\pi}{a}\right)\left(\frac{q\pi}{b}\right) \\
-2A_{mn}C_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{p\pi}{a}\right) + C_{mn}C_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right) \\
+ 2\Bigg(A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{p\pi}{a}\right)\left(\frac{q\pi}{b}\right) - A_{mn}B_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right) \\
- A_{mn}C_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{p\pi}{a}\right) + B_{mn}C_{pq}\left(\frac{p\pi}{a}\right)\left(\frac{q\pi}{b}\right)\Bigg) \\
+ A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{p\pi}{a}\right)\left(\frac{q\pi}{b}\right) \\
-2A_{mn}B_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right) + B_{mn}B_{pq}\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right)\Bigg]\Bigg][COS] \\
+ S_xB_{mn}B_{pq}cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} + S_yC_{mn}C_{pq}sin\frac{m\pi x}{a}cos\frac{n\pi y}{b}sin\frac{p\pi x}{a}cos\frac{q\pi y}{b}\Bigg]dxdy
\end{aligned}
\tag{A.2.5}
$$

$$
\begin{aligned}
W = \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{Q}\int_{y=0}^{b}\int_{x=0}^{a}\Bigg[N_xA_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{p\pi}{a}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} \\
+ N_yA_{mn}A_{pq}\left(\frac{n\pi}{b}\right)\left(\frac{q\pi}{b}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b} \\
+ 2N_{xy}A_{mn}A_{pq}\left(\frac{m\pi}{a}\right)\left(\frac{q\pi}{b}\right)cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}sin\frac{p\pi x}{a}cos\frac{q\pi y}{b}\Bigg]dxdy
\end{aligned}
\tag{A.2.6}
$$

The terms δ_{mp} and δ_{nq} are the Kronecker delta, these ensure that the integrals are only non-zero for terms where p equals m and q equals n. Similarly I_{mp} and I_{nq} are matrices which ensure that the integrals are only non zero when, for this integral, p is *not* equal to m and q is *not* equal to n.

Because all the integrals in the expression for the potential energy (except the shear load term, N_{xy}) are non-zero only when $m = p$ and $n = q$, we can replace p with m and q with n (except in the shear load term, N_{xy}) and then we get the final expressions for the strain energy and load potential as shown in Eqs. (4.2.14) and (4.2.15)

A.3 Integrals used in the potential energy

The integrals appearing in the potential energy expression are: Appearing in the terms with D_x, D_y and $D_x\nu_{yx} + D_y\nu_{xy}$:

$$\int_{y=0}^{b}\int_{x=0}^{a} sin\frac{m\pi x}{a}sin\frac{n\pi y}{b}sin\frac{p\pi x}{a}sin\frac{q\pi y}{b}dxdy$$

$$= \int_{x=0}^{a} sin\frac{m\pi x}{a}sin\frac{p\pi x}{a}dx \int_{y=0}^{b} sin\frac{n\pi y}{b}sin\frac{q\pi y}{b}dy = \begin{cases} \frac{a}{2} & \text{if } m = p \\ 0 & \text{if } m \neq p \end{cases} \cdot \begin{cases} \frac{b}{2} & \text{if } n = q \\ 0 & \text{if } n \neq q \end{cases}$$

$$= \begin{cases} \frac{ab}{4} & \text{if } m = p \text{ and } n = q \\ 0 & \text{otherwise} \end{cases} = \frac{ab}{4} \cdot \delta_{mp}\delta_{nq} \qquad \text{(A.3.1)}$$

Appearing in the terms with $D_{xy}/2$:

$$\int_{y=0}^{b}\int_{x=0}^{a} cos\frac{m\pi x}{a}cos\frac{n\pi y}{b}cos\frac{p\pi x}{a}cos\frac{q\pi y}{b}dxdy$$

$$= \int_{x=0}^{a} cos\frac{m\pi x}{a}cos\frac{p\pi x}{a}dx \int_{y=0}^{b} cos\frac{n\pi y}{b}cos\frac{q\pi y}{b}dy = \begin{cases} \frac{a}{2} & \text{if } m = p \\ 0 & \text{if } m \neq p \end{cases} \cdot \begin{cases} \frac{b}{2} & \text{if } n = q \\ 0 & \text{if } n \neq q \end{cases}$$

$$= \begin{cases} \frac{ab}{4} & \text{if } m = p \text{ and } n = q \\ 0 & \text{otherwise} \end{cases} = \frac{ab}{4} \cdot \delta_{mp}\delta_{nq} \qquad \text{(A.3.2)}$$

Appearing in the terms with S_x and N_x:

$$\int_{y=0}^{b}\int_{x=0}^{a} cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}cos\frac{p\pi x}{a}sin\frac{q\pi y}{b}dxdy$$

$$= \int_{x=0}^{a} cos\frac{m\pi x}{a}cos\frac{p\pi x}{a}dx \int_{y=0}^{b} sin\frac{n\pi y}{b}sin\frac{q\pi y}{b}dy = \begin{cases} \frac{a}{2} & \text{if } m = p \\ 0 & \text{if } m \neq p \end{cases} \cdot \begin{cases} \frac{b}{2} & \text{if } n = q \\ 0 & \text{if } n \neq q \end{cases}$$

$$= \begin{cases} \frac{ab}{4} & \text{if } m = p \text{ and } n = q \\ 0 & \text{otherwise} \end{cases} = \frac{ab}{4} \cdot \delta_{mp}\delta_{nq} \qquad \text{(A.3.3)}$$

Appearing in the terms with S_y and N_y:

$$\int_{y=0}^{b}\int_{x=0}^{a} sin\frac{m\pi x}{a}cos\frac{n\pi y}{b}sin\frac{p\pi x}{a}cos\frac{q\pi y}{b}dxdy$$

$$= \int_{x=0}^{a} sin\frac{m\pi x}{a}sin\frac{p\pi x}{a}dx \int_{y=0}^{b} cos\frac{n\pi y}{b}cos\frac{q\pi y}{b}dy = \begin{cases} \frac{a}{2} & \text{if } m = p \\ 0 & \text{if } m \neq p \end{cases} \cdot \begin{cases} \frac{b}{2} & \text{if } n = q \\ 0 & \text{if } n \neq q \end{cases}$$

$$= \begin{cases} \frac{ab}{4} & \text{if } m = p \text{ and } n = q \\ 0 & \text{otherwise} \end{cases} = \frac{ab}{4} \cdot \delta_{mp}\delta_{nq} \qquad \text{(A.3.4)}$$

Appearing in the term with N_{xy}:

$$\int_{y=0}^{b}\int_{x=0}^{a} cos\frac{m\pi x}{a}sin\frac{n\pi y}{b}sin\frac{p\pi x}{a}cos\frac{q\pi y}{a}dxdy$$

$$= \int_{x=0}^{a} cos\frac{m\pi x}{a}sin\frac{p\pi x}{a}dx \int_{y=0}^{b} sin\frac{n\pi y}{b}cos\frac{q\pi y}{b}dy$$

$$= \begin{cases} \frac{a[1-(-1)^m(-1)^p]p}{\pi(p^2-m^2)} & \text{if } m \neq p \\ 0 & \text{if } m = p \end{cases} \cdot \begin{cases} \frac{b[1-(-1)^n(-1)^q]n}{\pi(n^2-q^2)} & \text{if } n \neq q \\ 0 & \text{if } n = q \end{cases}$$

$$= \left[\frac{a(1-(-1)^m(-1)^p)p}{\pi(p^2-m^2)}\right]\left[\frac{b(1-(-1)^n(-1)^q)n}{\pi(n^2-q^2)}\right] I_{mp}I_{nq} \qquad \text{(A.3.5)}$$

Appendix B

Rayleigh-Ritz FORTRAN scripts

B.1 Main program file

```
PROGRAM Eigenproblem

  !Defines numeric precision
  USE kind_values,  only: wp => wp_swan

  REAL(wp) :: a, b, t1, t2, tc, Ef, Ec, nuf, nuc, k_rot1, k_rot2, k_rot3, theta, thetamax, thetamin
        k_rot4, N10, N20, N30, N10pre, N20pre, N30pre, lambda
  INTEGER iLambda, maxMa, maxNa, maxMb, maxNb, maxMc, maxNc, Itheta, analysisnumber, analysistype
  DOUBLE PRECISION,PARAMETER :: PI = 3.141592654

  DOUBLE PRECISION, DIMENSION(:), ALLOCATABLE :: LambdaArray


  !************************************************************************
  !* Plate properties                                                     *
  !************************************************************************
  a = 4190
  b = 2800
  t1 = 4
  t2 = 4
  tc = 25
  Ef = 208000
  Ec = 750
  nuf = 0.3
  nuc = 0.3


  !************************************************************************
  !* Boundary conditions (Large stiffness = clamped plate)                *
  !************************************************************************

  !***********Clamped*****************
  k_rot1 = 1000000000000000000
  k_rot2 = 1000000000000000000
  k_rot3 = 1000000000000000000
  k_rot4 = 1000000000000000000
```

```
!**********Simply supported*********
!k_rot1 = 0
!k_rot2 = 0
!k_rot3 = 0
!k_rot4 = 0

!************************************************************************
!* Number of the eigenvalue                                             *
!************************************************************************
iLambda = 1


!************************************************************************
!* Degrees of freedom                                                   *
!************************************************************************
maxMa= 20
maxNa= 20
maxMb= 20
maxNb= 20
maxMc= 20
maxNc= 20


iNumDof = maxMa*maxNa
kNumDof = maxMa*maxNa+maxMb*maxNb+maxMc*maxNc

!************************************************************************
!* Load conditions - analysistype                                       *
!************************************************************************

! Analysistype= 1 if single load combination
!       2 if interaction curve with constant N30
!       3 if interaction curve with constant N20
!       4 if interaction curve with constant N10

analysistype=1

IF (analysistype==1) THEN
  N10=33
  N20=0
  N30=0
  Call Eigenvalue(a, b, t1, t2, tc, Ef, Ec, nuf, nuc, &
      N10, N20, N30, N10pre, N20pre, N30pre, iNumDof, kNumDof, iLambda, maxMa, &
      maxNa, maxMb, maxNb, maxMc, maxNc, k_rot1, k_rot2, k_rot3, k_rot4, Itheta, lambda, analysis
  PRINT *,"RAYLEIGH_RITZ :"
  PRINT *, "LAMBDA :"
  PRINT *,lambda
  pause



ELSE IF (analysistype==2) THEN
  analysisnumber=16
  thetastep=PI/18.0
  theta=-3*PI/18.0

  allocate(LambdaArray(analysisnumber))
  DO Itheta=1, analysisnumber
    N10=cos(theta)*(t1+t2+tc)
    N20=sin(theta)*(t1+t2+tc)
    N30pre=0
    Call Eigenvalue(a, b, t1, t2, tc, Ef, Ec, nuf, nuc, &
```

```
      N10, N20, N30, N10pre, N20pre, N30pre, iNumDof, kNumDof, iLambda, maxMa, &
      maxNa, maxMb, maxNb, maxMc, maxNc, k_rot1, k_rot2, k_rot3, k_rot4, Itheta, lambda, analysis
    LambdaArray(Itheta)=lambda

    PRINT *,"Analysis number", Itheta
    PRINT *,"Eigenvalue",lambda
    theta=theta+thetastep

  END DO

  !**********************************************************************
  !* Print the eigenvector to a file                                    *
  !**********************************************************************
  OPEN(unit=1, access='sequential', status='replace', name='LambdaArray.csv')
    DO intI = 1 ,analysisnumber
      WRITE(1,*) LambdaArray(intI) ! Use format '(A1,F20.2)' instead of * if needed
    END DO
  CLOSE(1)
  !**********************************************************************
  !**********************************************************************



ELSE IF (analysistype==3) THEN
  !analysisnumber=15
  analysisnumber=25
  thetastep=PI/18.0
  theta=-12*PI/18.0

  allocate(LambdaArray(analysisnumber))
  DO Itheta=1, analysisnumber
    N10=cos(theta)*(t1+t2+tc)
    N30=sin(theta)*(t1+t2+tc)
    N20pre=267

    Call Eigenvalue(a, b, t1, t2, tc, Ef, Ec, nuf, nuc, &
      N10, N20, N30, N10pre, N20pre, N30pre, iNumDof, kNumDof, iLambda, maxMa, &
      maxNa, maxMb, maxNb, maxMc, maxNc, k_rot1, k_rot2, k_rot3, k_rot4, Itheta, lambda, analysis
    LambdaArray(Itheta)=lambda

    PRINT *,"Analysis number", Itheta
    PRINT *,"Eigenvalue",lambda
    theta=theta+thetastep

  END DO

  !**********************************************************************
  !* Print the eigenvector to a file                                    *
  !**********************************************************************
  OPEN(unit=1, access='sequential', status='replace', name='LambdaArray.csv')
    DO intI = 1 ,analysisnumber
      WRITE(1,*) LambdaArray(intI) ! Use format '(A1,F20.2)' instead of * if needed
    END DO
  CLOSE(1)
  !**********************************************************************
  !**********************************************************************



ELSE IF (analysistype==4) THEN
  analysisnumber=15
  thetastep=PI/18.0
```

```
    theta=-3*PI/18.0

    allocate(LambdaArray(analysisnumber))
    DO Itheta=1, analysisnumber
      N20=cos(theta)*(t1+t2+tc)
      N30=sin(theta)*(t1+t2+tc)
      N10pre=0
      Call Eigenvalue(a, b, t1, t2, tc, Ef, Ec, nuf, nuc, &
        N10, N20, N30, N10pre, N20pre, N30pre, iNumDof, kNumDof, iLambda, maxMa, &
        maxNa, maxMb, maxNb, maxMc, maxNc, k_rot1, k_rot2, k_rot3, k_rot4, Itheta, lambda, analysis
      LambdaArray(Itheta)=lambda

      PRINT *,"Analysis number", Itheta
      PRINT *,"Eigenvalue",lambda
      theta=theta+thetastep

    END DO

    !**********************************************************************
    !* Print the eigenvector to a file                                    *
    !**********************************************************************
    OPEN(unit=1, access='sequential', status='replace', name='LambdaArray.csv')
      DO intI = 1 ,analysisnumber
        WRITE(1,*) LambdaArray(intI) ! Use format '(A1,F20.2)' instead of * if needed
      END DO
    CLOSE(1)
    !**********************************************************************
    !**********************************************************************
  END IF

Pause
END PROGRAM
```

B.2 Plate stiffness matrices

```
SUBROUTINE Eigenvalue(a, b, t1, t2, tc, Ef, Ec, nuf, nuc, &
       N10, N20, N30, N10pre, N20pre, N30pre, iNumDof, kNumDof, iLambda, maxMa, &
       maxNa, maxMb, maxNb, maxMc, maxNc, k_rot1, k_rot2, k_rot3, k_rot4, &
       Itheta, lambda, analysistype)

  USE EigenproblemHandler

  IMPLICIT NONE

  DOUBLE PRECISION a, b, t1, t2, tc, Ef, Ec, nuf, nuc, Dx, Dy, Dxy, Gc, Sx, Sy, &
           darm, lambda, N10, N20, N30, N10pre, N20pre, N30pre, &
           k_rot1, k_rot2, k_rot3, k_rot4, xValue, yValue, &
           xArgument, yArgument

  INTEGER iLambda, maxMa, maxNa, maxMb, maxNb, maxMc, maxNc, iNumDof, &
           kNumDof, IJ, KL, analysistype

  INTEGER M, n, U, intI, intJ, intK, intL, intM, intN, intO, intP, intQ, intR, &
           INFOB1, INFOB2, INFOC1, INFOC2, IPIVC(maxMc*maxNc), &
           IPIVB(maxMa*maxNa), WORKB, WORKC, LWORKB, LWORKC, Itheta

  DOUBLE PRECISION,PARAMETER :: PI = 3.141592654
```

```
! INITIALISE THE STIFFNESS MATRICES AND THEIR DIMENSIONS
  DOUBLE PRECISION  KMAA(maxMa*maxNa,maxMa*maxNa), &
            KMBB(maxMb*maxNb,maxMb*maxNb), &
            KMCC(maxMc*maxNc,maxMc*maxNc), &
            KMAB(maxMa*maxNa,maxMb*maxNb), &
            KMAC(maxMa*maxNa,maxMc*maxNc), &
            KMBC(maxMb*maxNb,maxMc*maxNc), &
            KMBA(maxMb*maxNb,maxMa*maxNa), &
            KMCA(maxMc*maxNc,maxMa*maxNa), &
            KMCB(maxMc*maxNc,maxMb*maxNb), &
            KGAA(maxMa*maxNa,maxMa*maxNa), &
            KMPLATE(kNumDof,kNumDof), &
            KGPLATE(kNumDof,kNumDof), &
            KMCOND(maxMa*maxNa,maxMa*maxNa), &
            KGCOND(maxMa*maxNa,maxMa*maxNa), &
            KGPRE(maxMa*maxNa,maxMa*maxNa), &
            KMCCinv(maxMc*maxNc,maxMc*maxNc), &
            KMCCinverse(maxMc*maxNc,maxMc*maxNc), &
            TEST(maxMc*maxNc,maxMc*maxNc), &
            KMABCinv(maxMb*maxNb,maxMb*maxNb), &
            KMABC(maxMb*maxNb,maxMb*maxNb)

  DOUBLE PRECISION, DIMENSION(kNumDof,iLambda) :: EigenVec
  DOUBLE PRECISION, DIMENSION(iLambda) :: YMU
  DOUBLE PRECISION, DIMENSION(50,50) :: w_r

!**************** SANDWICH PLATE STIFFNESS COEFFICIENTS ****************
  Dx = Ec * tc**3 / (12 * (1 - nuc**2)) + Ef / (4 * (1 - nuf**2)) * &
      (t1 * (tc + t1)**2 + t2 * (tc + t2)**2)
  Dy = Ec * tc**3 / (12 * (1 - nuc**2)) + Ef / (4 * (1 - nuf**2)) * &
      (t1 * (tc + t1)**2 + t2 * (tc + t2)**2)
  Dxy = Ec * tc**3 / (12 * (1 + nuc)) + Ef / (4 * (1 + nuf)) * &
      (t1 * (tc + t1)**2 + t2 * (tc + t2)**2)
  darm = (t1/2)+(t2/2)+tc
  Gc = Ec/(2*(1+nuc))
  Sx = Gc/tc*darm**2
  Sy = Gc/tc*darm**2

  !PRINT *,"Dx:"
  !PRINT *,Dx
  !PRINT *,"Dy:"
  !PRINT *,Dy
  !PRINT *,"Dxy:"
  !PRINT *,Dxy

!***************** INITIALISE THE STIFFNESS MATRICES*******************
  KMAA = 0
  KMAB = 0
  KMAC = 0
  KMBB = 0
  KMBC = 0
  KMCC = 0
  KMBA = 0
  KMCA = 0
  KMCB = 0
  KGAA = 0
  KMPLATE = 0
  KGPLATE = 0
  KMCCinv = 0
  KMCCinverse=0
  KMABC = 0
  KMABCinv = 0
```

```
  KMCOND = 0
  KGCOND = 0
  TEST=0

!********************** COMPUTE THE AijAkl MATRIX **********************
  DO intI=1,maxMa;
    DO intJ=1,maxNa;
      DO intK = 1, maxMa
        DO intL = 1, maxNa
        IJ=maxNa*intI-maxNa+intJ
        KL=maxNa*intK-maxNa+intL
        IF (analysistype==1) THEN
          IF (intI==intK .and. intJ==intL) THEN
            KMAA(IJ,KL)=KMAA(IJ,KL)+(a*b/8)*(2*Dx*((intK*PI/a)**2*(intI*PI/a)**2)&
              +(Dx*Nuf+Dy*Nuf)*((intK*PI/a)**2*(intJ*PI/b)**2+(intI*PI/a)**2*(intL*PI/b)**2)&
              +2*Dy*((intL*PI/b)**2*(intJ*PI/b)**2)&
              +(2*Dxy*(2*(intI*PI/a)*(intJ*PI/b)*(intK*PI/a)*(intL*PI/b))))

            KGAA(IJ,KL)=KGAA(IJ,KL)-(a*b/8)*(2*N10*(intI*PI/a)&
                 *(intK*PI/a)+2*N20*(intJ*PI/b)*(intL*PI/b))
          END IF

          IF (intI/=intK .and. intJ/=intL) THEN
            KGAA(IJ,KL)=KGAA(IJ,KL)-N30*(a*(1-(-1)**intI*(-1)**intK)*intK)&
              /(PI*(intK**2-intI**2))&*(b*(1-(-1)**intJ*(-1)**intL)*intJ)&
              /(PI*(intJ**2-intL**2))*(intI*PI/a)*(intL*PI/b)&
              -N30*(a*(1-(-1)**intK*(-1)**intI)*intI)/(PI*(intI**2-intK**2))&
              *(b*(1-(-1)**intL*(-1)**intJ)*intL)&
              /(PI*(intL**2-intJ**2))*(intK*PI/a)*(intJ*PI/b)
          END IF
        ELSEIF (analysistype==2) THEN
          IF (intI==intK .and. intJ==intL) THEN
            KMAA(IJ,KL)=KMAA(IJ,KL) + (a*b/8)*(2*Dx*((intK*PI/a)**2*(intI*PI/a)**2)&
              +(Dx*Nuf+Dy*Nuf)*((intK*PI/a)**2*(intJ*PI/b)**2+(intI*PI/a)**2*(intL*PI/b)**2)&
              +2*Dy*((intL*PI/b)**2*(intJ*PI/b)**2)&
              +(2*Dxy*(2*(intI*PI/a)*(intJ*PI/b)*(intK*PI/a)*(intL*PI/b))))

            KGAA(IJ,KL) = KGAA(IJ,KL)-(a*b/8)*(2*N10*(intI*PI/a)&
              *(intK*PI/a)+2*N20*(intJ*PI/b)*(intL*PI/b))
          END IF

          IF (intI/=intK .and. intJ/=intL) THEN
            KMAA(IJ,KL)=KMAA(IJ,KL)-N30pre*(a*(1-(-1)**intI*(-1)**intK)*intK)&
              /(PI*(intK**2-intI**2))*(b*(1-(-1)**intJ*(-1)**intL)*intJ)&
              /(PI*(intJ**2-intL**2))*(intI*PI/a)*(intL*PI/b)&
              -N30pre*(a*(1-(-1)**intK*(-1)**intI)*intI)/(PI*(intI**2-intK**2))&
              *(b*(1-(-1)**intL*(-1)**intJ)*intL)&
              /(PI*(intL**2-intJ**2))*(intK*PI/a)*(intJ*PI/b)
          END IF
        ELSEIF (analysistype==3) THEN
          IF (intI==intK .and. intJ==intL) THEN
            KMAA(IJ,KL)=KMAA(IJ,KL)+(a*b/8)*(2*Dx*((intK*PI/a)**2*(intI*PI/a)**2)&
              +(Dx*Nuf+Dy*Nuf)*((intK*PI/a)**2*(intJ*PI/b)**2+(intI*PI/a)**2*(intL*PI/b)**2)&
              +2*Dy*((intL*PI/b)**2*(intJ*PI/b)**2)&
              +(2*Dxy*(2*(intI*PI/a)*(intJ*PI/b)*(intK*PI/a)*(intL*PI/b))))


            KMAA(IJ,KL) = KMAA(IJ,KL)-(a*b/8)*2*N20pre*(intJ*PI/b)*(intL*PI/b)
            KGAA(IJ,KL) = KGAA(IJ,KL)-(a*b/8)*2*N10*(intI*PI/a)*(intK*PI/a)
          END IF

          IF (intI/=intK .and. intJ/=intL) THEN
```

```
            KGAA(IJ,KL)=KGAA(IJ,KL)-N30*(a*(1-(-1)**intI*(-1)**intK)*intK)&
              /(PI*(intK**2-intI**2))*(b*(1-(-1)**intJ*(-1)**intL)*intJ)&
              /(PI*(intJ**2-intL**2))*(intI*PI/a)*(intL*PI/b)&
              -N30*(a*(1-(-1)**intK*(-1)**intI)*intI)/(PI*(intI**2-intK**2))&
              *(b*(1-(-1)**intL*(-1)**intJ)*intL)&
              /(PI*(intL**2-intJ**2))*(intK*PI/a)*(intJ*PI/b)
          END IF
        ELSEIF (analysistype==4) THEN
          IF (intI==intK .and. intJ==intL) THEN
            KMAA(IJ,KL)=KMAA(IJ,KL)+(a*b/8)*(2*Dx*((intK*PI/a)**2*(intI*PI/a)**2)&
              +(Dx*Nuf+Dy*Nuf)*((intK*PI/a)**2&*(intJ*PI/b)**2+(intI*PI/a)**2*(intL*PI/b)**2)&
              +2*Dy*((intL*PI/b)**2*(intJ*PI/b)**2)&
              +(2*Dxy*(2*(intI*PI/a)*(intJ*PI/b)*(intK*PI/a)*(intL*PI/b))))


            KMAA(IJ,KL) = KMAA(IJ,KL)-(a*b/8)*2*N10pre*(intI*PI/a)*(intK*PI/a)
            KGAA(IJ,KL) = KGAA(IJ,KL)-(a*b/8)*2*N20*(intJ*PI/b)*(intL*PI/b)
          END IF

          IF (intI/=intK .and. intJ/=intL) THEN
            KGAA(IJ,KL) = KGAA(IJ,KL) &
              -N30*(a*(1-(-1)**intI*(-1)**intK)*intK)/(PI*(intK**2-intI**2))*(b*(1-(-1)**intJ&
              *(-1)**intL)*intJ)/(PI*(intJ**2-intL**2))*(intI*PI/a)*(intL*PI/b)&
              -N30*(a*(1-(-1)**intK*(-1)**intI)*intI)/(PI*(intI**2-intK**2))*(b*(1-(-1)**intL&
              *(-1)**intJ)*intL)/(PI*(intL**2-intJ**2))*(intK*PI/a)*(intJ*PI/b)
          END IF
        END IF

        !****************** ELASTIC SPRINGS AT THE BOUNDARY ****************
        IF (intL==intJ) THEN
          KMAA(IJ,KL)=KMAA(IJ,KL)+(1.0/2)*b*(intI*PI/a)*(intK*PI/a)*k_rot1

          KMAA(IJ,KL)=KMAA(IJ,KL)&
              +(1.0/2)*b*(intI*PI/a)*(intK*PI/a)*k_rot2*cos(intK*PI)*cos(intI*PI)
        END IF

        IF (intK==intI) THEN
          KMAA(IJ,KL)=KMAA(IJ,KL)+(1.0/2)*a*(intJ*PI/b)*(intL*PI/b)*k_rot3

          KMAA(IJ,KL)=KMAA(IJ,KL)&
              +(1.0/2)*a*(intJ*PI/b)*(intL*PI/b)*k_rot4*cos(intL*PI)*cos(intJ*PI)
        END IF

        END DO
      END DO
    END DO
  END DO

!********************** COMPUTE THE BijBkl MATRIX **********************
  DO intI=1,maxMb;
    DO intJ=1,maxNb;
      DO intK = 1, maxMb
        DO intL = 1, maxNb
        IJ=maxNb*intI-maxNb+intJ
        KL=maxNb*intK-maxNb+intL
        IF (intI==intK .and. intJ==intL) THEN
          KMBB(IJ,KL) = KMBB(IJ,KL)&
            +(a*b/4)*(Dx*(intI*PI/a)*(intK*PI/a)+Dxy/2*(intJ*PI/b)*(intL*PI/b)+Sx)
        END IF

        !****************** ELASTIC SPRINGS AT THE BOUNDARY ********************
        IF (intL==intJ) THEN
```

```
          KMBB(IJ,KL)=KMBB(IJ,KL)+(1.0/2)*b*k_rot1

          KMBB(IJ,KL)=KMBB(IJ,KL)+(1.0/2)*b*k_rot2*cos(intK*PI)*cos(intI*PI)
        END IF

        END DO
      END DO
    END DO
  END DO

!********************** COMPUTE THE CijCkl MATRIX **********************
DO intI=1,maxMc;
  DO intJ=1,maxNc;
    DO intK = 1, maxMc
      DO intL = 1, maxNc
        IJ=maxNc*intI-maxNc+intJ
        KL=maxNc*intK-maxNc+intL
        IF (intI==intK .and. intJ==intL) THEN
          KMCC(IJ,KL) = KMCC(IJ,KL)&
            +(a*b/4)*(Dy*(intJ*PI/b)*(intL*PI/b)+Dxy/2*(intI*PI/a)*(intK*PI/a)+Sy)
        END IF

        !****************** ELASTIC SPRINGS AT THE BOUNDARY ********************
        IF (intK==intI) THEN
          KMCC(IJ,KL)=KMCC(IJ,KL)+0.5*a*k_rot3

          KMCC(IJ,KL)=KMCC(IJ,KL)+0.5*a*k_rot4*cos(intL*PI)*cos(intJ*PI)
        END IF

      END DO
    END DO
  END DO
END DO

!********************** COMPUTE THE AijBkl MATRIX **********************
  DO intI=1,maxMa;
    DO intJ=1,maxNa;
      DO intK = 1, maxMb
        DO intL = 1, maxNb
        IJ=maxNa*intI-maxNa+intJ
        KL=maxNb*intK-maxNb+intL
        IF (intI==intK .and. intJ==intL) THEN
          KMAB(IJ,KL)=-(KMAB(IJ,KL)+(a*b/8)*(2*Dx*(intI*PI/a)**2*(intK*PI/a)&
            +(Dx*nuf+Dy*nuf)*(intK*PI/a)*(intJ*PI/b)**2&
            +2*Dxy*(intI*PI/a)*(intJ*PI/b)*(intL*PI/b)))
        END IF

        !****************** ELASTIC SPRINGS AT THE BOUNDARY ********************
        IF (intL==intJ) THEN
        KMAB(IJ,KL)=KMAB(IJ,KL)-(1.0/2)*b*(intI*PI/a)*k_rot1

        KMAB(IJ,KL)=KMAB(IJ,KL)-(1.0/2)*b*(intI*PI/a)*k_rot2*cos(intK*PI)*cos(intI*PI)
        END IF


        END DO
      END DO
    END DO
  END DO

!********************** COMPUTE THE AijCkl MATRIX **********************
  DO intI=1,maxMa;
```

```
    DO intJ=1,maxNa;
      DO intK = 1, maxMc
        DO intL = 1, maxNc
          IJ=maxNa*intI-maxNa+intJ
          KL=maxNc*intK-maxNc+intL
          IF (intI==intK .and. intJ==intL) THEN
            KMAC(IJ,KL)=-(KMAC(IJ,KL)+(a*b/8)*(2*Dy*(intJ*PI/b)**2*(intL*PI/b)&
              +(Dx*nuf+Dy*nuf)*(intI*PI/a)**2*(intL*PI/b)&
              +2*Dxy*(intI*PI/a)*(intJ*PI/b)*(intK*PI/a)))
          END IF


          !****************** ELASTIC SPRINGS AT THE BOUNDARY ********************
          IF (intK==intI) THEN
          KMAC(IJ,KL)=KMAC(IJ,KL)-(1.0/2)*a*(intJ*PI/b)*k_rot3

          KMAC(IJ,KL)=KMAC(IJ,KL)-(1.0/2)*a*(intJ*PI/b)*k_rot4*cos(intJ*PI)*cos(intL*PI)
          END IF
        END DO
      END DO
    END DO
  END DO

!********************** COMPUTE THE BijCkl MATRIX **********************
  DO intI=1,maxMb;
    DO intJ=1,maxNb;
      DO intK = 1, maxMc
        DO intL = 1, maxNc
          IJ=maxNb*intI-maxNb+intJ
          KL=maxNc*intK-maxNc+intL
          IF (intI==intK .and. intJ==intL) THEN
            KMBC(IJ,KL)=KMBC(IJ,KL)&
            +(a*b/8)*((Dx*nuf+Dy*nuf)*(intI*PI/a)*(intL*PI/b)+Dxy*(intK*PI/a)*(intL*PI/b))
          END IF
        END DO
      END DO
    END DO
  END DO

!************************ OPTION ONE: FULL SYSTEM ************************


!******************* COMPUTE THE TRANSPOSED MATRICES *******************
  KMBA=Transpose(KMAB)
  KMCA=Transpose(KMAC)
  KMCB=Transpose(KMBC)
!********************** COMPUTE THE BijBkl MATRIX **********************
!***************** ASSEMBLE THE PLATE STIFFNESS MATRIX *****************
!
!     MATRIX(from row:to row, from column:to column)
!
!***********************************************************************
  KMPLATE(1:maxMa*maxNa, 1:maxMa*maxNa) = &
  KMPLATE(1:maxMa*maxNa, 1:maxMa*maxNa) + KMAA

  KMPLATE(maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) = &
  KMPLATE(maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) + KMBB

  KMPLATE(maxMa*maxNa+maxMb*MaxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc,&
    maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) = &
  KMPLATE(maxMa*maxNa+maxMb*MaxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc,&
    maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) + KMCC
```

```
  KMPLATE(1:maxMa*maxNa, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) = &
  KMPLATE(1:maxMa*maxNa, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) + KMAB

  KMPLATE(1:maxMa*maxNa, maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) = &
  KMPLATE(1:maxMa*maxNa, maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) + KMAC

  KMPLATE(maxMa*MaxNa+1:maxMa*maxNa+maxMb*maxNb,&
    maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) = &
  KMPLATE(maxMa*MaxNa+1:maxMa*maxNa+maxMb*maxNb,&
    maxMa*maxNa+maxMb*maxNb+1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc) + KMBC

  KMPLATE(maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb, 1:maxMa*maxNa) = &
  KMPLATE(maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb, 1:maxMa*maxNa) + KMBA

  KMPLATE(maxMa*maxNa+maxMb*maxNb&
  +1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc, 1:maxMa*maxNa) = &
  KMPLATE(maxMa*maxNa+maxMb*maxNb&
  +1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc, 1:maxMa*maxNa) + KMCA

  KMPLATE(maxMa*maxNa+maxMb*maxNb&
  +1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) = &
  KMPLATE(maxMa*maxNa+maxMb*maxNb&
  +1:maxMa*maxNa+maxMb*maxNb+maxMc*maxNc, maxMa*maxNa+1:maxMa*maxNa+maxMb*maxNb) + KMCB

  KGPLATE(1:maxMa*maxNa, 1:maxMa*maxNa) = &
  KGPLATE(1:maxMa*maxNa, 1:maxMa*maxNa) + KGAA


  !****************************************************************
  !* Solving the eigenvalue problem
  !****************************************************************
  CALL GiveInPutData(iLambda , 'M' , kNumDof, M, -KGPLATE, KMPLATE)
  CALL GetOutPutData(YMU , Eigenvec )
  lambda = 1 / YMU(1)
  !****************************************************************

!************************* END OF OPTION ONE *************************


!******************* OPTION TWO: CONDITIONED SYSTEM ******************
!
! KMCCinv = KMCC
! CALL DPOTRF('U',maxNc*maxNc, KMCCinv, maxNc*maxNc,INFOC1)
! CALL DPOTRI('U',maxNc*maxNc, KMCCinv, maxNc*maxNc,INFOC1)
!
! CALL DGETRF(maxMc*maxNc, maxMc*maxNc, KMCCinv, maxMc*maxNc, IPIVC, INFOC1)
! CALL DGETRI(maxMc*maxNc, KMCCinv, maxMc*maxNc, IPIVC, WORKC, maxMc*maxNc, INFOC2)
!
! TEST=matmul(KMCCinv,KMCC)
!
! CALL DGETRF(maxMc*maxNc, maxMc*maxNc, KMCCinv, maxMc*maxNc, IPIVC, INFOC1)
! CALL DGETRI(maxMc*maxNc, KMCCinv, maxMc*maxNc, IPIVC, WORKC, maxMc*maxNc, INFOC2)
!
!
! CALL DGETRF(maxMb*maxNa, maxMa*maxNa, KMABCinv, maxMa*maxNa, IPIVB, INFOB1)
! CALL DGETRI(maxMb*maxNa, KMABCinv, maxMa*maxNa, IPIVB, WORKB, maxMa*maxNa, INFOB2)
!
!
! KMCCinv = KMCC
! CALL DTRTRI('U', 'N', maxMc*maxNc, KMCCinv, maxMc*maxNc, INFOC1)
! KMABCinv=KMBB - matmul(KMBC, matmul(KMCCinv,KMCB))
```

```
!
!
! CALL DTRTRI('U', 'N', maxMc*maxNc, KMCCinv, maxMc*maxNc, INFOC1)
!
!
! KMABCinv=KMBB - matmul(KMBC, matmul(KMCCinv,KMCB))
!
! CALL DTRTRI('U', 'N', maxMa*maxNa, KMABCinv, maxMa*maxNa, INFOB1)
!
! KMCOND=(KMAA-matmul(KMAC,matmul(KMCCinv,KMCA)))&
!      -matmul(KMAB-matmul(KMAC,matmul(KMCCinv,KMCB)),&
!      matmul(KMABCinv,(KMBA - matmul(KMBC,matmul(KMCCinv,KMCA)))))
!
! KGCOND=KGAA
!
! !*********************************************************************
! !* Solving the eigenvalue problem
! !*********************************************************************
! CALL GiveInPutData(iLambda , 'M' , iNumDof, M, -KGCOND, KMCOND)
! CALL GetOutPutData(YMU , Eigenvec )
! lambda = 1 / YMU(1)
! !*********************************************************************
! !*********************************************************************
!
!**************** END OF OPTION TWO: CONDITIONED SYSTEM ****************

  !*********************************************************************
  !* The displacement matrix (radial displacement)
  !*********************************************************************
  DO intM = 1 ,50
    DO intN = 1 ,50
      xValue = (intN - 1)*a/49
      yValue = (intM - 1)*b/49
      intR = 0
      DO intP = 1 ,maxMa
        DO intQ = 1 ,maxNa
          xArgument = PI*intP*xValue/a
          yArgument = PI*intQ*yValue/b
          intR = intR + 1
          w_r(intN,intM)=w_r(intN,intM)&
            +Eigenvec(intR,1)*sin(xArgument)*sin(yArgument)
        END DO
      END DO
    END DO
  END DO
  !*********************************************************************

  !*********************************************************************
  !* Print the eigenvector to a file
  !*********************************************************************
  !OPEN(unit=11, access='sequential', status='replace', name='Eigenmode.dat')
  ! DO intI = 1 ,maxMa*maxNa
  !         WRITE(11,'(A1,F20.15)') ',',Eigenvec(intI,1)
  ! END DO
  !CLOSE(11)
  !*********************************************************************
END SUBROUTINE Eigenvalue
```

Appendix C

ABAQUS PYTHON scripts

C.1 MPC Slider script

```
# Run syntax:
# abaqus viewer noGui=filename.py
import os
from odbAccess import *
from abaqusConstants import *
from abaqus import *
from visualization import *
from numpy import *

Analysisname='Buckle'
# Specify the edges of the plate, and the interface coordinates
a=4190
b=2800
platebottom=0
platemiddle=16.5
platetop=33

# Specify the file to write the slider commands
f = file('SliderThroughoutPlate.txt','w')

# Specify odb
odbName=Analysisname+'.odb'

#odbName=Analysisname+'.odb'
odb=openOdb(odbName)

#Thickness of the entire plate
tp=33

#Angi step
StepName='Buckle'
#angi hvilket increment i angitt steg
FrameNo=1
OutputC=odb.steps[StepName].frames[FrameNo].fieldOutputs['COORD']
f.write(' *'+'MPC' '\n')
for Value1 in OutputC.values:
  x1=Value1.data[0]
```

```
  y1=Value1.data[1]
  z1=Value1.data[2]
        # At the lower boundary where y=0:
        if z1==platemiddle:
                NodeA=Value1.nodeLabel
                for Value1 in OutputC.values:
                        x2=Value1.data[0]
                        y2=Value1.data[1]
                        z2=Value1.data[2]
                        if z2==0:
                                NodeB=Value1.nodeLabel
                                if x2==x1 and y2==y1:
                                        for Value1 in OutputC.values:
                                                x3=Value1.data[0]
                                                y3=Value1.data[1]
                                                z3=Value1.data[2]
                                                NodeP=Value1.nodeLabel
                                                if x3==x1 and y3==y1 and z3!=z1
                                                   and z3!=z2:


                                                        %
                                                        NodeP+',Part-1-1.'+'%s'
                                                        %
                                                        NodeA+',Part-1-1.'+'%s'
                                                        % NodeB+'\n')


f.close()

# '\n''SLIDER,''%s' % NodeA #+ NodeP,NodeA,NodeB + '\n'
```

C.2 Script for interaction curves input files

```
   #-----------------------------------------------------------------------------------------
#       Write INPUT FILES for LINEAR and NONLINEAR analysis
#       ABAQUS Solid models, takes two base input files, changes the load
   combinations
#       Ole J. Hareide, UIO/DNV 2012

   #-----------------------------------------------------------------------------------------

import os, os.path, math, string, time

GO_input_base_linear  =  "Buckle.inp"
GO_input_base_nonlinear = "NonLinear.inp"
GO_out_base     =  "NonLinear"
GO_out_folder   =  "N30=0 SPS imp200"
GO_script       =  "./" + GO_out_folder + "/runme.bat"

Pi=3.14159265358979323846
starttheta=-3*Pi/18
thetastep=Pi/18
analysisnumber=16
analysistype=1
jobs = []
ln=0
```

```
if os.path.isdir( ( "./" + GO_out_folder ) ) != 1:
  os.mkdir( ( "./" + GO_out_folder ) )

fscr = open( GO_script,  'w' )

   #-------------------------------------------------------------------------------------------
#               Write input for eigenvalue analysis

   #-------------------------------------------------------------------------------------------
if analysistype==1:
        theta=starttheta
        for i in range(0, analysisnumber):
                angle=theta*180/Pi
                last1=math.cos(theta)
                last2=math.sin(theta)

                GO_output_buckle = ( ( "Buckle_%04.1f_%04.2f_%04.2f" )
                        % ( angle, last1, last2 ) )

                finp = open( ( "./" + GO_input_base_linear ),   'r' )

                fout = open( ( "./" + GO_out_folder + "/" + GO_output_buckle +
                   ".inp" ), 'w' )

                fscr.write( "call abaqus job=%s interactive\n" %
                   GO_output_buckle )

                for line in finp:
                        if "BottomSurface, P" in line:
                                fout.write(line.replace(line,("BottomSurface, P,
                                   "+"%08.6f\n") %last2))
                        elif "LeftSurface, P" in line:
                                fout.write(line.replace(line,("LeftSurface, P,
                                   "+"%08.6f\n") %last1))
                        elif "RightSurface, P" in line:
                                fout.write(line.replace(line,("RightSurface, P,
                                   "+"%08.6f\n") %last1))
                        elif "TopSurface, P" in line:
                                fout.write(line.replace(line,("TopSurface, P,
                                   "+"%08.6f\n") %last2))
                        else:
                                fout.write(line)
                theta=theta+thetastep
                ln += 1
                finp.close()
                fout.close()

   #-------------------------------------------------------------------------------------------
#               Write input for non-linear analysis

   #-------------------------------------------------------------------------------------------
if analysistype==1:
        theta=starttheta
        for i in range(0, analysisnumber):
                angle=theta*180/Pi
                last1=math.cos(theta)
                last2=math.sin(theta)

                GO_output_nonlinear = ( ( "NonLinear_%04.1f_%04.2f_%04.2f" )
                        % ( angle, last1, last2 ) )

                finp = open( ( "./" + GO_input_base_nonlinear),   'r' )
```

```
                fout = open( ( "./" + GO_out_folder + "/" + GO_output_nonlinear
                   + ".inp" ), 'w' )

                fscr.write( "call abaqus job=%s interactive\n" %
                   GO_output_nonlinear )

                for line in finp:
                        if "** MATERIALS" in line:
                                fout.write(line+'\n'+'**
                                   IMPERFECTION'+'\n'+'*IMPERFECTION,
                                   FILE='+("Buckle_%04.1f_%04.2f_%04.2f")
                        % ( angle, last1, last2 )+', STEP=1' +'\n'+'1, 14.0\n')
                        elif "** STEP: Buckle" in line:
                                fout.write(line.replace(line, "** STEP:
                                   NonLinearRiks\n"))
                        elif "*Step, name=Buckle, perturbation" in line:
                                fout.write(line.replace(line, "*Step,
                                   name=NonLinearRiks, nlgeom=YES, inc=20\n"))
                        elif "*Buckle" in line:
                                fout.write(line.replace(line, "*Static, riks \n
                                   1., 1., 1e-05, , ,"))
                        elif "BottomSurface, P" in line:
                                fout.write(line.replace(line,("BottomSurface, P,
                                   "+"%08.6f\n") %last2))
                        elif "LeftSurface, P" in line:
                                fout.write(line.replace(line,("LeftSurface, P,
                                   "+"%08.6f\n") %last1))
                        elif "RightSurface, P" in line:
                                fout.write(line.replace(line,("RightSurface, P,
                                   "+"%08.6f\n") %last1))
                        elif "TopSurface, P" in line:
                                fout.write(line.replace(line,("TopSurface, P,
                                   "+"%08.6f\n") %last2))
                        else:
                                fout.write(line)
                theta=theta+thetastep
                ln += 1
                finp.close()
                fout.close()
fscr.close()
#time.sleep(10)
#os.startfile("C:/ABAQUS/UlsteinNonLinearSS/N30=0/runme.bat")
```