Diploma Thesis

Fictitious Domain Approach for Optimizing Stability Boundaries of Plates with Cutouts

submitted in satisfaction of the requirements for the degree of
Diplom-Ingenieur
of the TU Wien, Faculty of Mechanical and Industrial Engineering

---

Diplomarbeit

Fiktiver Gebietsansatz für die Optimierung der Stabilitätsgrenzen von Platten mit Löchern

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines
Diplom-Ingenieurs
eingereicht an der Technischen Universität Wien, Fakultät für Maschinenwesen und Betriebswissenschaften

von

Robert Duy, BSc

Matr.Nr.: 01325476

unter der Anleitung und Betreuung von

Privatdoz. Mag. Dr. **Yury Vetyukov**

Institut für Mechanik und Mechatronik
Forschungsbereich Mechanik fester Körper
Technische Universität Wien
Getreidemarkt 9/E325, 1060 Wien, Österreich

Dr. **Sascha Eisenträger**

School of Civil and Environmental Engineering
Research Area Structural Engineering
University of New South Wales Sydney
Kensington Campus, 2052 NSW, Australia

Wien, im Februar 2021

Robert Duy, BSc

Ich nehme zur Kenntnis, dass ich zur Drucklegung dieser Arbeit unter der Bezeichnung

DIPLOMARBEIT

nur mit Bewilligung der Prüfungskommission berechtigt bin.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass die vorliegende Arbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen von mir selbstständig erstellt wurde. Alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, sind in dieser Arbeit genannt und aufgelistet. Die aus den Quellen wörtlich entnommenen Stellen sind als solche kenntlich gemacht.

Das Thema dieser Arbeit wurde von mir bisher weder im In- noch Ausland einer Beurteilerin/einem Beurteiler zur Begutachtung in irgendeiner Form als Prüfungsarbeit vorgelegt. Diese Arbeit stimmt mit der von den Begutachterinnen/Begutachtern beurteilten Arbeit überein.

Ich nehme zur Kenntnis, dass die vorgelegte Arbeit mit geeigneten und dem derzeitigen Stand der Technik entsprechenden Mitteln (Plagiat-Erkennungssoftware) elektronisch-technisch überprüft wird. Dies stellt einerseits sicher, dass bei der Erstellung der vorgelegten Arbeit die hohen Qualitätsvorgaben im Rahmen der geltenden Regeln zur Sicherung guter wissenschaftlicher Praxis (*Code of Conduct*) an der TU Wien eingehalten wurden. Zum anderen werden durch einen Abgleich mit anderen studentischen Abschlussarbeiten Verletzungen meines persönlichen Urheberrechts vermieden.

Wien, am 12.02.2021 Robert Duy, BSc

Kurzfassung

Die vorliegende Diplomarbeit hat das Ziel der Weiterentwicklung und Verbesserung der numerischen Methoden zur Berechnung strukturmechanischer Probleme. Als besonderes Beispiel aus der Strukturmechanik wird das Stabilitätsproblem des Beulens von dünnen Platten herangezogen. Je nach Aufgabenstellung stehen für einfache Probleme analytische Lösungen bereit; jedenfalls aber werden numerische Methoden wie die *Finite Elemente Methode* (FEM) mit großem Erfolg eingesetzt um auch Probleme mit komplexeren Geometrien, Lastfällen und Randbedingungen behandeln zu können.

Die Behandlung von Platten mit Löchern allerdings ist immer noch mit gewissen Schwierigkeiten verbunden. Zum einen exisitieren dafür nur semi-analytische Lösungsansätze, zum anderen ist die Diskretisierung des Problems im Rahmen der Finite Elemente Methode mit einigem Aufwand verbunden. Dies gilt besonders bei wiederholten Analysen, die sich nur durch einzelne, bestimmte Parameter unterscheiden, wie zum Beipiel der Position des Loches.

Die *Finite Zellen Methode* (FCM) als Weiterentwicklung der FEM wird in dieser Arbeit derart eingesetzt, dass der Aufwand zur Erzeugung konformer Diskretisierungen entfällt, wenn die Position des Loches in der Platte geändert wird. Die FCM beruht auf der Idee, dass die Geometrie des eigentlichen Problems in ein größeres Gebiet eingebettet wird und dieses Gebiet wiederum in sogenannte physikalische und fiktive Gebiete unterteilt wird. Diese Einbettung begründet den Namen „Fiktiver Gebietsansatz“. Durch die folgende nicht-geometiekonforme Diskretisierung des Problems müssen diskontinuierliche Integranden behandelt werden. Die fordernste Aufgabe zur Lösung des Problems ist nun nicht mehr die geometriekonforme Diskretisierung wie bei der FEM, sondern die Berechnung der Systemmatrizen, die die genaue Integration der diskontinuierlichen Integranden erfordert.

Die Einführung eines nicht-geometriekonformen Netzes, das in diesem Fall rein kartesisch ist, erlaubt auch die Verwendung von Ansatzfunktionen, die aufgrund gewisser Nachteile in FEM-Berechnungen kaum mehr Anwendungen finden. In dieser Arbeit werden die sogenannten *Bogner-Fox-Schmit* (BFS) Ansatzfunktionen verwendet. Die ausgezeichneten Konvergenzeigenschaften der damit definierten BFS-Elemente werden in der Berechnung des ebenen Spannungszustands, des Platten- und des Beulproblems ausgenutzt; sie sorgen, gemeinsam mit der Effizienz der FCM, für ein schnelles und leicht adaptierbares Lösen des Beulproblems für Platten.

Abschließend wird diese neue Methode eingesetzt um die Stabilitätsgrenzen von Platten mit Löchern zu optimieren. Dafür wird ein Plattenproblem vorgestellt, bei dem die Position eines Loches mit bekanntem Radius gesucht wird, die die kritische Beullast maximiert. Entgegen der anfänglichen Intuition zeigt sich, dass die Einführung von Löchern die Beullast nicht immer verringert, sondern sie im Vergleich zur Vollplatte auch erhöhen kann!

Abstract

The focus of the present master's thesis is on further developing and improving numerical methods for their application in structural mechanics. In particular, the stability problem of buckling of thin plates is considered. Depending on the geometry of the structure and its loading conditions, analytical solutions exist only for simple problems. For more complex problems however, numerical methods like the *Finite Elements Method* (FEM) are applied with great success. Doing so, it is also possible to deal with complex geometries, loading, and boundary conditions.

The treatment of plates with cutouts is, however, still associated with certain difficulties. There only exist semi-analytical solutions, and, more importantly, the discretization of the problem is elaborate; even more so in the case of repeated analyses which only differ in few parameters, for example the position of the cutout.

In this thesis the *Finite Cell Method* (FCM), as an extension of traditional FEM, is applied in a way that the mesh generation burden in FEM is alleviated when the position of the hole changes. FCM is based on embedding the original problem geometry in a larger one, the so-called *extended domain*. The extended domain is then again split in the *physical* and the *fictitious* domains, establishing the name "Fictitious Domain Approach" for this method. The subsequent non-geometry conforming discretization introduces discontinuous integrands. Thus, the difficulty in solving this problem now no longer lies in the geometry-conforming discretization and preparation of the problem, but in the computation of the system matrices, which requires precise evaluation of the discontinuous integrands.

The introduction of a non-geometry conforming discretization, which is in this case strictly Cartesian, allows the application of shape functions, which are rarely applied today due to certain limitations. In this thesis, the so-called *Bogner-Fox-Schmit* (BFS) shape functions are used. Finite elements based on these functions show an excellent convergence behaviour; along with the efficiency of FCM this ensures the fast and easy-to-modify solution of the buckling problem of plates.

Finally, this newly developed method is applied to optimize the stability boundaries of plates with cutouts. A plate problem is introduced and the position of a hole is of interest, for which the critical buckling load is maximized. Counterintuitively, it is shown that the introduction of cutouts does not always result in a decreased critical load, but that in fact it can be increased compared to a plate without holes!

Contents

List of Figures

List of Symbols

γ	Ratio in buckling strength	h	Plate thickness
$\boldsymbol{u}, \underset{\sim}{\boldsymbol{u}}$	Vector of displacements	$\boldsymbol{m}$	Tensor of moments
u	Displacement in 1-direction	$\boldsymbol{\kappa}$	Tensor of curvatures
v	Displacement in 2-direction	D	Plate bending stiffness
w	Displacement in 3-direction	$\mathbf{D}$	Plate bending stiffness matrix
$\boldsymbol{\sigma}$, $\underset{\sim}{\boldsymbol{\sigma}}$	Cauchy stress	$\boldsymbol{I}$	Identity tensor
$\boldsymbol{\varepsilon}$, $\underset{\sim}{\boldsymbol{\varepsilon}}$	Green-Lagrange strain	$tr(\cdot)$	Trace operator
$\mathbf{C}$, $^4\underset{\sim}{\boldsymbol{C}}$	Elasticity matrix/tensor	$\delta W^{(i)}$	Virtual work of internal forces
∇	Nabla operator	$\delta W^{(e)}$	Virtual work of external forces
$(\cdot)^{\mathrm{T}}$	Transpose operator	$\boldsymbol{\mathcal{L}}$	Double gradient operator
$\partial(\cdot)/\partial x_i$	Partial derivative with respect to x_i	$\boldsymbol{\mathscr{L}}$	Gradient operator
E	Elastic modulus	$\boldsymbol{\Sigma}$	Extended stress matrix
ν	Poison's ratio	$\mathbf{K}_{\mathrm{P}}$	Plate bending stiffness matrix
Ω	Domain	$\mathbf{K}_{\mathrm{G}}$	Plate initial stress stiffness matrix
$\mathcal{B}$	Bilinear form	λ	Eigenvalue / load factor
$\mathcal{F}$	Linear form	$\boldsymbol{\Lambda}$	Vector of eigenvalues
$\delta(\cdot)$	Virtual expression	$\boldsymbol{\Phi}$	Vector of eigenmodes
$\mathbf{L}$	Strain-displacement operator	Ω_e	Extended domain
$\boldsymbol{b}$	Vector of body forces	Ω_{phys}	Physical domain
$\boldsymbol{t}_{\mathrm{N}}$	Vector of traction forces	Ω_{fict}	Fictitious domain
Γ_{N}	Neumann boundary	k	Level of refinement
$\mathbf{N}(\cdot)$	Matrix of shape functions	$\alpha(\boldsymbol{x})$	Indicator function
ξ, η, ζ	Parametric coordinates	$\mathbf{K}^c$	Stiffness matrix of cell c
$\det(\cdot)$	Determinant operator	$\mathbf{K}^{(c,sc)}$	Stiffness matrix of sub-cell (c, sc)
$\mathbf{J}$	Jacobi matrix	r, s	Parametric coordinates
$\boldsymbol{k}_e$	Elemental stiffness matrix	$\mathcal{S}$	Strain energy
$\mathbf{K}$	Global stiffness matrix	n_x	Number of elements along edge
$\mathbf{U}$	Global displacement vector	$\mathcal{A}$	Area
$\boldsymbol{f}_e$	Elemental force vector	δ	Stepsize
$\mathbf{F}$	Global force vector		

List of Abbreviations

BFS	Bogner-Fox-Schmit	FEM	Finite Element Method
DOF	Degree(s) of Freedom	IP	Integration Point
FCM	Finite Cell Method	LC	Load Case

Chapter 1

Introduction

The theory of elastic stability of slender structures like beams and plates always has been an important feature in structural mechanics. Euler, as early as in the 1740ies, studied the buckling of thin, long columns under axial compression and such created the still valid basis of stability theory within structural mechanics [1]. In the course of time, many authors since contributed to this topic; Timoshenko and Gere [2] published their standard reference work "Theory of Elastic Stability" for dealing with such problems in 1961. Modern publications like "Buckling and Postbuckling of Beams, Plates, and Shells" by M. Eslami [3] also include chapters on thermal or thermo-electrical buckling and buckling of modern engineering materials, like functionally graded materials or materials exhibiting piezo-electric properties.

Today, different approaches to treat stability problems in computational structural mechanics are available. The first one is the linear eigenvalue analysis, which can be traced back to Euler. A generalized eigenvalue problem has to be solved and the resulting lowest positive eigenvalue corresponds to the factor, by which the applied load has to be multiplied to bring the structure into buckling. Its key advantages are the underlying simplicity and its short computation time, which are for example applied in [4], where the buckling of elastic structures under tensile loads is analyzed. One of the most important disadvantage however is, that the buckling load is usually overestimated when using this approach. This is because of its linear nature, it cannot take into account geometric or material non-linearity nor does it consider imperfections of any kind, which real structures show. To overcome these disadvantages non-linear buckling analysis is also available. Its main feature is that it divides the loading process into steps and applies the load incrementally. This way, also geometrically non-linear analysis can be performed and non-linear material behavior can be used, as well as imperfections considered. An example may be a metallic shell structure that undergoes large deformation and plastic yielding before buckling. Another example can be found in [5], where shell buckling of imperfect cylindrical structures is studied. An additional reference on the comparison of linear and non-linear buckling analysis for thin shells is [6].

In this thesis, the simplicity of linear eigenvalue analysis is utilized to evaluate the buckling strength at all admissible positions within the problem geometry. The short computation time is an important characteristic here, as it allows an assessment of the buckling strength as a function of the hole position in reasonable time. It may be advisable, however, to analyze an engineering structure, with the hole position already defined, also with a non-linear approach. The method presented in this thesis is rather suitable for preliminary design, where the accuracy of linear buckling analysis is perfectly sufficient.

Recently, Gracia and Rammerstorfer [7] revisited the subject of buckling of isotropic, homogeneous, linear elastic plates with cutouts, loaded by in-plane edge forces. Specifically, they presented an approach to identify how the position of these cutouts influences the buckling strength. After introducing the common procedure for determining buckling loads by using a Ritz-ansatz in combination with the so-called Rayleigh quotient, the effect of a cutout in the plate is studied. The concept of "sensor holes", which are infinitesimal small, virtual, circular holes cut into the plate, is introduced. A way is then found to apply these sensor holes such that an indicator function can be created, where the effect of the hole position can directly be seen when plotted. An example of such a plot is displayed in Fig. 1.1, where γ is shown over the plate geometry. γ is the ratio between the buckling strength of the plate with a sensor hole and the buckling strength of the unperforated plate; bright areas signal high values of γ and the white lines shown are isolines, where $\gamma = 1$. It can already be seen that there are areas with γ values bigger than 1. This of course means, that the introduction of a sensor hole in these areas increases buckling strength. It is now assumed, that the introduction of finite holes at favorable positions (which means where $\gamma > 1$) also increases buckling strength. This is verified for multiple positions and load cases with FEM. For example, the introduction of a hole with radius $\frac{r}{a} = 0.055$ at the dimensionless position (0.9, 0.5) (dashed circle in Fig. 1.1) increases the critical buckling force by nearly 1% for a certain load case. Cutting the same hole at an unfavorable position, like at the middle of the plate, the buckling strength is decreased by up to 5%. The effect of the hole may be small, but from an engineering viewpoint it is very desirable to place holes, if they are necessary, at favorable positions. It may even be possible to increase buckling strength and to reduce structural mass at the same time!

However, γ-plots created with this concept should be interpreted as suggestions for suitable positions only. Real cutouts may very well behave differently. Therefore, it is always

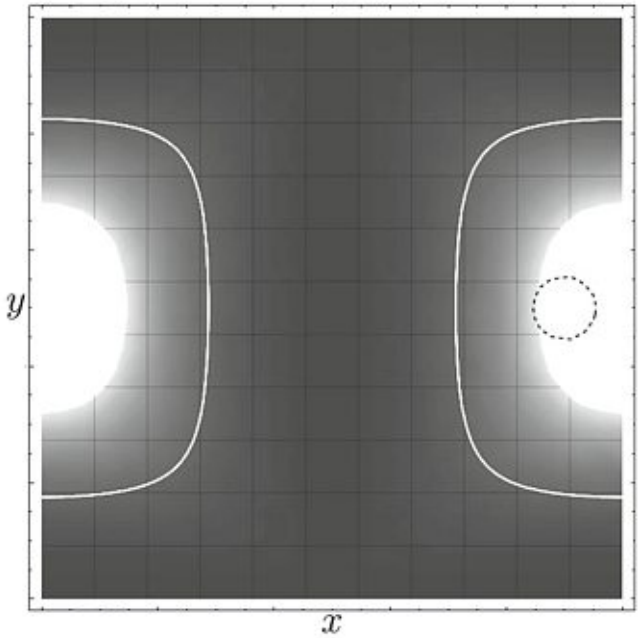


Fig. 1.1: Exemplary plot showing buckling strength as a function of the hole position [7].

necessary to perform additional analysis. In most commercial FEM software the analysis of a plate with a cutout is an easy task; it becomes cumbersome however to repeat analyses with different positions for the hole. This is because it is necessary to create a new mesh for each hole position, which usually is the most time-consuming task in an FEM analysis.

This thesis now introduces the possibility to create plots similar to the γ-plots. They, however, show not only suggested positions for cutouts, but actually the ratio in buckling strength for the chosen hole geometry. The limitations of commercial FEM software in this respect is overcome by the application of FCM. FCM is introduced in more detail in Section 2.4; it allows to use the same mesh for all hole positions, omitting the repeated mesh creation needed for FEM.

After this detailed description of the problem at hand and the goals of this thesis, the following sections will briefly introduce the numerical methods used and the theoretical background needed. This will aid in a better understanding for the rest of the thesis. Section 2.1 will revise linear elasticity and the state of plane stress specifically. It is needed to compute the stress distribution of the plate, which is then used in buckling analysis. Section 2.2 introduces the needed basics of plate theory and Sections 2.3 and 2.4 will briefly sketch the theory of FEM and FCM respectively.

Chapter 3 elaborates in detail how a provided FCM-MATLAB code for plates was extended, which intermediate steps were performed and how the code was verified. Chapter 4 finally discusses the numerical results from the optimization and Chapter 5 summarizes the implementation process and the obtained results, as well as all lessons learned. Finally, an outlook with considerations for possible future work is provided.

Please note that all governing equations of linear elasticity and plate theory are introduced in component-free tensor notation. Invariant objects can be identified by a tilde underneath the symbol. For example, $\underset{\sim}{\boldsymbol{\sigma}}$ represents the second-order stress tensor and ${}^{4}\underset{\sim}{\boldsymbol{C}}$ is the fourth-order elasticity tensor.

At some point, a change in notation is performed, and the equations are adapted to Voigt-Matrix notation in a Cartesian basis. This way, the computer implementation for subsequent numeric simulation becomes more transparent. Variables printed in bold, without a tilde underneath, represent the matrices of components of respective vectorial or tensorial quantities. The second-order stress tensor is now represented with $\boldsymbol{\sigma} = (\sigma_{11}\ \sigma_{22}\ \sigma_{33}\ \tau_{23}\ \tau_{13}\ \tau_{12})^{\mathrm{T}}$ and the fourth-order elasticity tensor can be denoted with the 6x6 matrix $\boldsymbol{C}$.

Chapter 2

Theoretical Background and Methods

2.1 Plane Stress Problem of Linear Elasticity

In this thesis, only the plane problem of linear elasticity is considered, which for thin bodies like plates is known to follow the plane stress assumptions. Therefore, this section will shortly introduce the plane stress problem, derive a weak formulation and introduce a suitable discretization.

The starting point is to define a suitable strain measure in terms of the displacements $\underset{\sim}{\boldsymbol{u}}$, which consist of the components u in 1-direction and v in 2-direction for the state of plane stress. In linear elasticity, under the assumption of small displacements and small strains, this measure is the linearized Green-Lagrange strain tensor $\underset{\sim}{\boldsymbol{\varepsilon}}$, see for example [8, 9]. $\underset{\sim}{\boldsymbol{\varepsilon}}$ is the symmetric part of the displacement gradient tensor and defined as

$$\underset{\sim}{\boldsymbol{\varepsilon}} = \frac{1}{2}[\nabla \underset{\sim}{\boldsymbol{u}} + (\nabla \underset{\sim}{\boldsymbol{u}})^{\mathrm{T}}], \tag{2.1}$$

where ∇ is the in-plane Nabla operator, given by

$$\nabla = \boldsymbol{e}_x \frac{\partial}{\partial x} + \boldsymbol{e}_y \frac{\partial}{\partial y}, \tag{2.2}$$

with unit vectors $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ in 1- and 2-direction.

Next, a relation between the Cauchy stress tensor $\underset{\sim}{\boldsymbol{\sigma}}$ and the strain tensor $\underset{\sim}{\boldsymbol{\varepsilon}}$ is needed, which is called a constitutive law and for HOOKEAN solids denoted as

$$\underset{\sim}{\boldsymbol{\sigma}} = {}^4\underset{\sim}{\mathbf{C}} \cdot\cdot \, \underset{\sim}{\boldsymbol{\varepsilon}}, \tag{2.3}$$

where ${}^4\underset{\sim}{\mathbf{C}}$ is the elasticity tensor. Now, the relation Eq. (2.3) is denoted in matrix notation as:

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon}. \tag{2.4}$$

A state of plane stress typically occurs in thin structures, that are loaded by in-plane forces only. The plane stress condition is introduced as a stress state, where all out-of-plane stress components vanish as $\sigma_{33} = \tau_{23} = \tau_{13} = 0$. Due to Poisson's effect, the strain component ε_{33} results as a non-zero term. It is, however, not explicitly included in the following Eq. (2.6), but can be determined from the resulting in-plane stress components. In Voigt notation, $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ are 3x1 column vectors and $\mathbf{C}$ is a 3x3 matrix.

In the case of isotropic material behavior, to which this thesis is restricted, $\mathbf{C}$ reads as

$$\mathbf{C} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}, \tag{2.5}$$

with the material properties Young's modulus E and Poisson's ratio ν. Eq. (2.4) can now be written as

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \tau_{12} \end{bmatrix} = \frac{E}{(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{bmatrix}. \tag{2.6}$$

Based on the principle of virtual work the governing equations for linear elasticity on a domain Ω can be stated in a weak formulation as the equality of a bilinear functional $\mathcal{B}(\boldsymbol{u}, \delta\boldsymbol{u})$ and a linear functional $\mathcal{F}(\delta\boldsymbol{u})$, see for example [10],

$$\mathcal{B}(\boldsymbol{u}, \delta\boldsymbol{u}) = \mathcal{F}(\delta\boldsymbol{u}), \tag{2.7}$$

in which $\boldsymbol{u}$ is the displacement field and $\delta\boldsymbol{u}$ a virtual displacement. The bilinear form is

$$\mathcal{B}(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_\Omega [\mathbf{L}\delta\boldsymbol{u}]^{\mathrm{T}} \mathbf{C} [\mathbf{L}\boldsymbol{u}] \, \mathrm{d}\Omega, \tag{2.8}$$

and the linear form is denoted as

$$\mathcal{F}(\delta\boldsymbol{u}) = \int_\Omega \delta\boldsymbol{u}^{\mathrm{T}} \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_{\mathrm{N}}} \delta\boldsymbol{u}^{\mathrm{T}} \boldsymbol{t}_{\mathrm{N}} \, \mathrm{d}\Gamma_{\mathrm{N}}, \tag{2.9}$$

where the standard strain-displacement operator $\mathbf{L}$, the vector of volume loads $\boldsymbol{b}$ and prescribed tractions $\boldsymbol{t}_{\mathrm{N}}$ at Neumann boundaries Γ_{N} are introduced. The linear differential operator $\mathbf{L}$ is given in Eq. (2.10) as

$$\mathbf{L} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix}, \tag{2.10}$$

and allows the computation of the strains by applying the operator $\mathbf{L}$ to the displacements $\boldsymbol{u}$ as in Eq. (2.11):

$$\boldsymbol{\varepsilon} = \mathbf{L}\boldsymbol{u}. \tag{2.11}$$

The displacement field $\boldsymbol{u}$ and the virtual displacement $\delta\boldsymbol{u}$ are approximated based on a Bubnov-Galerkin approach, applying the same shape functions, and in terms of generalized degrees of freedom (DOF) at fixed, discrete points:

$$\boldsymbol{u} = \mathbf{N}_{\mathrm{G}}(\boldsymbol{x})\mathbf{U}_{\mathrm{G}}, \tag{2.12}$$

$$\delta\boldsymbol{u} = \mathbf{N}_{\mathrm{G}}(\boldsymbol{x})\delta\mathbf{U}_{\mathrm{G}}. \tag{2.13}$$

$\mathbf{N}_{\mathrm{G}}(\boldsymbol{x})$ hereby denotes the matrix of global shape functions, which must be valid for the complete domain Ω, and is evaluated at $\boldsymbol{x}$; $\mathbf{U}_{\mathrm{G}}$ is the vector of generalized degrees of freedom. It often consists of the displacement field itself, but, depending on the used formulation and its interpolation scheme, it may also include derivatives. This is usually the case for beam, plate or shell problems because of the required continuity of the approximation [11].

The domain Ω is now discretized by dividing it in a finite number of elements, which are defined via their nodes. An exemplary mesh can be seen in Fig. 2.1, where a semicircular structure is discretized with different numbers of quadrilateral and triangular elements. The area integrals for both the bilinear form $\mathcal{B}$ and the linear form $\mathcal{F}$ are now divided into integrals over these finite elements. This means that both are assembled from $\mathcal{B}_e$ and $\mathcal{F}_e$, which each finite element contributes. $\mathcal{B}_e$ can be expressed in terms of $\mathbf{U}_e$ and $\boldsymbol{k}_e$; $\mathcal{F}_e$ is defined in terms of $\mathbf{U}_e$ and $\boldsymbol{f}_e$, as in Eqs. (2.14) and (2.15). $\mathbf{U}_e$ are the nodal degrees of freedom of a specific finite element. $\boldsymbol{k}_e$ is the so-called elemental stiffness matrix, $\boldsymbol{f}_e$ is the elemental force vector:

$$\mathcal{B}_e = \delta\mathbf{U}_e^{\mathrm{T}}\boldsymbol{k}_e\mathbf{U}_e, \tag{2.14}$$

$$\mathcal{F}_e = \delta\mathbf{U}_e^{\mathrm{T}}\boldsymbol{f}_e. \tag{2.15}$$

A particular advantage of this approach is, that globally valid shape functions $\mathbf{N}_{\mathrm{G}}(\boldsymbol{x})$ are no longer necessary. Instead, local shape functions $\mathbf{N}(\boldsymbol{\xi})$, or simply $\mathbf{N}$, are introduced, which are easier to obtain. Additionally, these local shape functions can be defined in terms of parametric coordinates ξ and η, where $\xi, \eta \in [-1; 1]$. Defining shape functions in the global coordinates $\boldsymbol{x}$ is impractical as we would need to define new shape functions for each element depending on its shape.

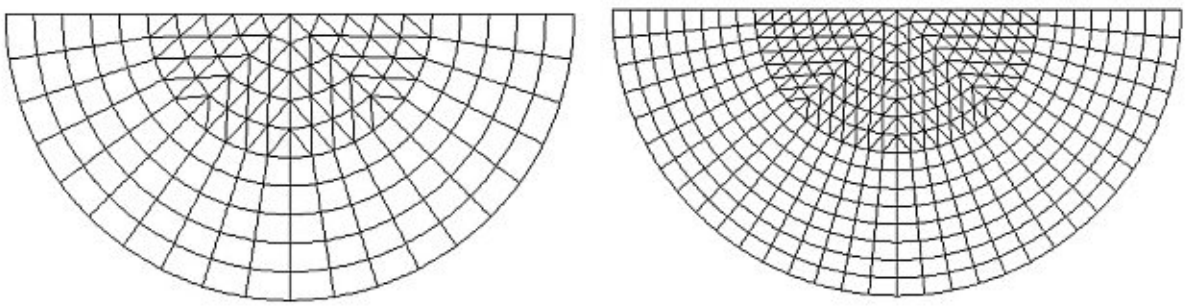


Fig. 2.1: Discretization of a semicircular structure with different numbers of quadrilateral and triangular elements, the total number is 200 on the left and 512 on the right. [12].

By defining the shape functions parametrically, they only need to be defined for a reference element, greatly simplifying the following numerical integration. However, a mapping between the original element geometry and the reference element geometry now needs to be introduced. $\mathbf{J}$ is the so-called Jacobian matrix of this mapping, defined through

$$\mathbf{J} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}, \tag{2.16}$$

or, in component notation, as

$$\mathbf{J} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix}. \tag{2.17}$$

Similar to the approximation of $\boldsymbol{u}$ and $\delta\boldsymbol{u}$ in Eqs. (2.12) and (2.13), inside a finite element the displacement $\boldsymbol{u}_e$ can be approximated with local shape functions $\mathbf{N}$ in terms of the nodal degrees of freedom $\mathbf{U}_e$:

$$\boldsymbol{u}_e = \mathbf{N}\mathbf{U}_e, \tag{2.18}$$

$$\delta\boldsymbol{u}_e = \mathbf{N}\delta\mathbf{U}_e. \tag{2.19}$$

Identity (2.7) now must hold for all admissible virtual displacements $\delta\boldsymbol{u}$. After incorporating the discretization from Eqs. (2.14) and (2.15) and the Ansatz of Eqs. (2.18) and (2.19) this results in an expression for the elemental stiffness matrix from $\mathcal{B}_e(\mathbf{U}_e, \delta\mathbf{U}_e)$ and a right-hand side force vector from $\mathcal{F}_e(\delta\mathbf{U}_e)$:

$$\boldsymbol{k}_e = \int_{-1}^{1}\int_{-1}^{1} [\mathbf{L}\mathbf{N}^{\mathrm{T}}]\mathbf{C}[\mathbf{L}\mathbf{N}] \det(\mathbf{J}) \,\mathrm{d}\xi\mathrm{d}\eta, \tag{2.20}$$

$$\boldsymbol{f}_e = \int_{-1}^{1}\int_{-1}^{1} \mathbf{N}^{\mathrm{T}}\boldsymbol{b} \,\mathrm{d}\xi\mathrm{d}\eta + \int_{-1}^{1} \mathbf{N}^{\mathrm{T}}\boldsymbol{t}_{\mathrm{N}} \,\mathrm{d}\zeta. \tag{2.21}$$

In Eq. (2.21) the element's Neumann boundary Γ_{N} is parameterized in terms of $\zeta \in [-1; 1]$. It is assumed that this part of the boundary, with external forces $\boldsymbol{t}_{\mathrm{N}}$ acting, corresponds to the coordinate line ζ.

The global stiffness matrix $\mathbf{K}$ and the global force vector $\mathbf{F}$ can now be obtained by assembling all element entities. Inside the global stiffness matrix $\mathbf{K}$, each elemental stiffness matrix $\boldsymbol{k}_e$ is put at its appropriate "location" according to its degrees of freedom [13]. The same is true for the assembly of the global force vector $\mathbf{F}$. This results in the algebraic global equation system

$$\mathbf{K}\mathbf{U} = \mathbf{F}, \tag{2.22}$$

with the unknown global degrees of freedom vector $\mathbf{U}$ as solution.

2.2 Plate Theory

In this section, a short summary of the KIRCHHOFF-LOVE plate theory is given. It strongly follows the notation used in [9]. The describing equations are again stated in a weak formulation, for which finally a discretization suitable for FEM is introduced.

A plate is a three-dimensional solid body with one of its dimensions, the thickness h, being much smaller than the other two. It is loaded such that only out-of-plane deflections w in $\boldsymbol{z}$-direction occur, to be differentiated from a membrane, which is also a thin structure, but loaded such that only in-plane displacements u and v occur. For both of them, the describing equations are formulated for their mid-surface. In Fig. 2.2 such a plate is depicted and its midsurface especially highlighted.

In its reference configuration there is zero curvature in the plate's mid-surface. Following the KIRCHHOFF-LOVE plate theory, it is assumed that in its deformed configuration, straight lines normal to the mid-surface remain straight and remain normal to the mid-surface. In other words, this means shear deformations are neglected, which is a permissible assumption for thin plates. It is also assumed that the thickness h is constant and does not change during deformation. Assuming small displacements and small strains, as in Section 2.1, a geometrically linear theory follows. Additionally, it is noted again that this thesis is restricted to isotropic material behavior.

The principle of virtual work for a plate can be written in terms of the tensor of curvatures $\underset{\sim}{\boldsymbol{\kappa}}$ and the tensor of moments $\underset{\sim}{\boldsymbol{m}}$. Both are in-plane tensors, considered in the mid-surface of the plate. Following [9], the elastic relation is

$$\underset{\sim}{\boldsymbol{m}} = D\nu \underset{\sim}{\mathbf{I}} tr\underset{\sim}{\boldsymbol{\kappa}} + D(1-\nu)\underset{\sim}{\boldsymbol{\kappa}}, \tag{2.23}$$

where $\underset{\sim}{\mathbf{I}}$ is the identity tensor, tr is the trace operator and D is the plate bending stiffness, defined as

$$D = \frac{Eh^3}{12(1-\nu^2)}. \tag{2.24}$$

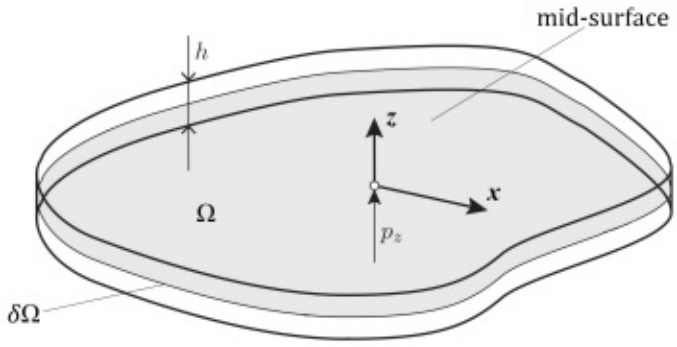


Fig. 2.2: A plate of thickness h, its highlighted mid-surface and an admissible load p_z. Adapted from [9].

From the kinematic relations it follows that $\underset{\sim}{\boldsymbol{\kappa}}$ can solely be expressed in terms of the out-of-plane displacement w, applying the double gradient,

$$\underset{\sim}{\boldsymbol{\kappa}} = \nabla\nabla w. \tag{2.25}$$

The principle of virtual work reads

$$\delta W^{(e)} - \delta W^{(i)} = 0, \tag{2.26}$$

where $\delta W^{(i)}$ is the virtual work done by internal forces and $\delta W^{(e)}$ is the virtual work done by external forces. In this thesis admissible external forces for plates are point loads and distributed loads p_z normal to the mid-surface only. $\delta W^{(i)}$ is expressed in terms of the tensor of moments $\underset{\sim}{\boldsymbol{m}}$ and the tensor of virtual curvatures $\delta\underset{\sim}{\boldsymbol{\kappa}}$, and $\delta W^{(e)}$ can be written in terms of the virtual deflection δw:

$$\delta W^{(i)} = \int_\Omega \underset{\sim}{\boldsymbol{m}} \cdot\cdot\, \delta\underset{\sim}{\boldsymbol{\kappa}} \, \mathrm{d}\Omega, \tag{2.27}$$

$$\delta W^{(e)} = \int_\Omega p_z \delta w \, \mathrm{d}\Omega. \tag{2.28}$$

Here, no boundary terms must be taken into account, as all plates analyzed in the following have their displacement w restrained at the boundaries. Additionally, no external moments act on these boundaries, and their respective contribution to the principle of virtual work is dropped.

After introducing Voigt-Matrix notation, the bilinear part $\mathcal{B}(w, \delta w)$ of the weak formulation now follows from $\delta W^{(i)}$ as

$$\mathcal{B}(w, \delta w) = \int_\Omega (\boldsymbol{\mathcal{L}}\delta w)^{\mathrm{T}} \mathbf{D} (\boldsymbol{\mathcal{L}} w) \, \mathrm{d}\Omega. \tag{2.29}$$

$\boldsymbol{\mathcal{L}}$ is a linear differential operator by which the matrix of curvature $\boldsymbol{\kappa}$ may be expressed in terms of w, as introduced in Eq. (2.25). Similar to the standard strain-displacement operator $\mathbf{L}$ known from linear elasticity, it can be defined as

$$\boldsymbol{\mathcal{L}} = \begin{bmatrix} \dfrac{\partial^2}{\partial x^2} \\ \dfrac{\partial^2}{\partial y^2} \\ \dfrac{\partial^2}{\partial x \partial y} \end{bmatrix}. \tag{2.30}$$

$\mathbf{D}$ is the plate stiffness matrix in Voigt notation,

$$\mathbf{D} = \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}. \tag{2.31}$$

The linear part $\mathcal{F}(\delta w)$ for the defined admissible forces follows from $\delta W^{(e)}$ as

$$\mathcal{F}(\delta w) = \int_\Omega p_z \delta w \, \mathrm{d}\Omega. \tag{2.32}$$

The domain Ω is again discretized and w and δw are again approximated with the same shape functions in the sense of a Bubnov-Galerkin procedure and expressed in terms of nodal values for each element. Following common practice in FEM-notation, the designations $\mathbf{N}, \mathbf{K}, \mathbf{U}$ and $\mathbf{F}$ are reused but must strictly be distinguished from terms introduced in Section 2.1:

$$w_e = \mathbf{N}\mathbf{U}_e, \tag{2.33}$$

$$\delta w_e = \mathbf{N}\delta\mathbf{U}_e. \tag{2.34}$$

With the same arguments as for the expressions shown in Eqs. (2.20) and (2.21), the elemental stiffness matrix and the elemental force vector are obtained as

$$\boldsymbol{k}_e = \int_{-1}^{1}\int_{-1}^{1} [\boldsymbol{\mathcal{L}}\mathbf{N}^{\mathrm{T}}]\mathbf{D}[\boldsymbol{\mathcal{L}}\mathbf{N}] \, \det\mathbf{J} \, \mathrm{d}\xi\mathrm{d}\eta, \tag{2.35}$$

$$\boldsymbol{f}_e = \int_{-1}^{1}\int_{-1}^{1} \mathbf{N}^{\mathrm{T}} p_z \, \mathrm{d}\xi\mathrm{d}\eta, \tag{2.36}$$

and again an algebraic global system of equations follows after the correct assembly of the elemental entities:

$$\mathbf{K}\mathbf{U} = \mathbf{F}. \tag{2.37}$$

2.3 Additional Aspects of the Finite Element Method

This section will briefly describe some additional aspects of the Finite Element Method. All of these are important for the following implementation, especially as the Finite Cell Method can be seen as an extension of FEM. This section does not claim any completeness, but elaborates selected topics. A thorough and detailed presentation of FEM can be found in the wide body of literature, for example in [13, 14].

2.3.1 Shape Functions

As introduced earlier, a discretization of the continuous problem is chosen, in which the displacement field is approximated in terms of nodal values and shape functions. It is usually not feasible to use shape functions covering the complete domain Ω. Instead it is reasonable and useful to define them in finite subsections of Ω only, leading to the notion of finite elements. The reason to do so is that the shape functions must fulfill certain requirements to be applicable and valid. Most importantly, they must comply with all kinematic boundary conditions. Finding functions doing so over the complete domain Ω may be possible for small problems, otherwise the requirements can only be fulfilled easily when restricted to the subsection of finite elements.

A big variety of shape functions has been successfully applied to all kinds of problems, ranging from simple linear interpolation schemes to high-order polynomial methods. They

must be chosen to fulfill not only the general requirement introduced above, but also those inherently implied by the problem to be solved. In plate theory – in contrast to linear elasticity – such an additional requirement is C^1-continuity of the primary variable, which is necessary due to the presence of second derivatives of the displacement w in the curvature definition. The continuity conditions between elements now have to be imposed not only on w, but also on its derivatives (thus C^1-continuity), ensuring a continuous strain measure. In a nutshell, this ensures that the plate remains a continuous body in which no kinks occur [11].

There exist multiple possibilities to deal with this stringent requirement for plate problems, among which are methods that independently interpolate rotations Φ and the displacement w, imposing the continuity additionally only at discrete points (as done in numerous shell element formulations in the commercial FEM-package ABAQUS [15]); or the introduction of Lagrange multiplier, both to avoid C^1-continuous shape functions, which are difficult to obtain. In fact, as stated in [11], "it is thus impossible to specify simple polynomial expressions for shape functions ensuring full compatibility when only w and its slopes are prescribed at corner nodes" due to consequential inconsistencies in the second cross-derivative $\frac{\partial^2 w}{\partial x \partial y}$ at nodes. One possibility to overcome this difficulty was first proposed by Bogner, Fox and Schmit in [16], where they added the second cross-derivative $\frac{\partial^2 w}{\partial x \partial y}$ as an additional nodal degree of freedom. This is permissible, however, restrictions on the structure of the mesh now follow, as continuous coordinate lines across element boundaries are required. An assembly of rectangular elements, as used in this thesis, fulfills this in any case, an extension to meshes with arbitrary topology is in general not permissible. Another possible application of this interpolation scheme is for shell problems, where mesh structures as displayed in Fig. 2.3 fulfill the requirement of continuous coordinate lines. Other than that, the "firm analytical background, simplicity of implementation, absence of rotational degrees of freedom, constant mass matrix and regular convergence" are particular advantages of this approximation, both for linear and nonlinear applications, as stated in the detailed presentation of this numerical scheme in [17].

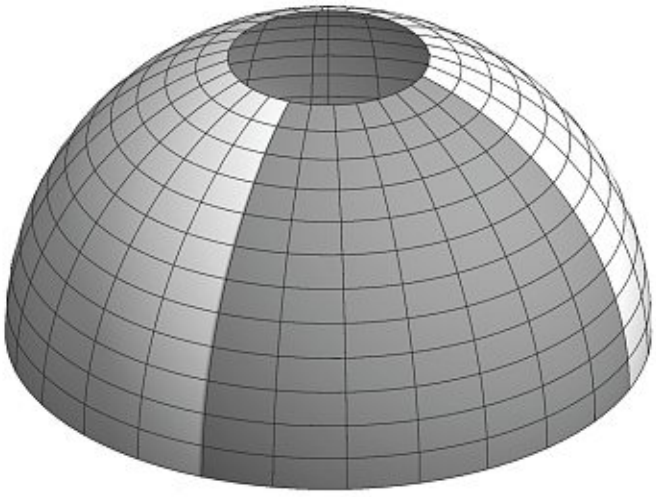


Fig. 2.3: Plot of a hemispherical dome mesh, that is analyed using BFS shell elements in [17].

This interpolation technique has not found widespread use besides academia, mainly due to the restrictions on mesh structure and because other element formulation turned out to be more flexible. In the context of FCM however, BFS-elements, as elements using this numerical scheme will be called here, may be a perfect fit for the discretization of the extended domain. In this thesis especially so, as the extended domain is of rectangular shape and discretized by square-shaped elements. The adaption to the physical geometry is performed with FCM, elegantly bypassing BFS-elements' limitations.

2.3.2 BFS-Element Formulation

Closely following [17], a reference BFS-element shall be formulated, however no longer using differential geometrical notation. Instead, the Cartesian pair of coordinates $(\xi, \eta) \in [-1; 1]$ is used. Denoting the four nodes of a reference element with i, j, k and l and nodal coordinates as in Table 2.1, the corresponding set of shape functions for node i can be stated in the form of Eq. (2.38).

$$\begin{aligned} N^{i,1} &= \frac{1}{16}\Big((\xi-1)^2(\eta-1)^2(\xi+2)(\eta+2)\Big) \\ N^{i,2} &= \frac{1}{16}\Big((\xi-1)^2(\eta-1)^2(\xi+1)(\eta+2)\Big) \\ N^{i,3} &= \frac{1}{16}\Big((\xi-1)^2(\eta-1)^2(\xi+2)(\eta+1)\Big) \\ N^{i,4} &= \frac{1}{16}\Big((\xi-1)^2(\eta-1)^2(\xi+1)(\eta+1)\Big) \end{aligned} \tag{2.38}$$

The shape functions belonging to the other three nodes can be obtained by substitution in Eq. (2.38). For node j, the substitution is $\xi \to (\xi - 1)$; for node k, it is $\eta \to (\eta - 1)$; and for node l both substitutions are performed. This set of shape functions can be derived from standard 1D-Hermite polynomials, which are widely used for beam elements. By using a tensor product formulation of these polynomials, the stated BFS-shape functions can be constructed, as for example demonstrated in [18]. Doing so, the requirement of interelement C^1-continuity is fulfilled analytically. An additional property is that they can represent any bicubic polynomial exactly, as 16 coefficients are needed to do so.

For practical application, the needed shape functions are available in MATLAB functions, for which the necessary derivatives were pre-calculated. After passing on a pair of coordinates (ξ, η), the shape function values are returned at this position. In linear elasticity, the first derivatives of the shape functions are needed to form the strain-displacement operator $\mathbf{L}$ introduced in Eq. (2.8). For plate theory, the second derivatives are needed to form the curvature-displacement operator $\boldsymbol{\mathcal{L}}$, which corresponds to the double gradient introduced in Eq. (2.25.) For the stability problem introduced in the following chapter, again first derivatives are needed, forming $\boldsymbol{\mathcal{L}}$, the gradient operator.

Table 2.1: Nodal coordinates of the reference element.

	i	j	k	l
(ξ, η)	$(-1, -1)$	$(1, -1)$	$(1, 1)$	$(-1, 1)$

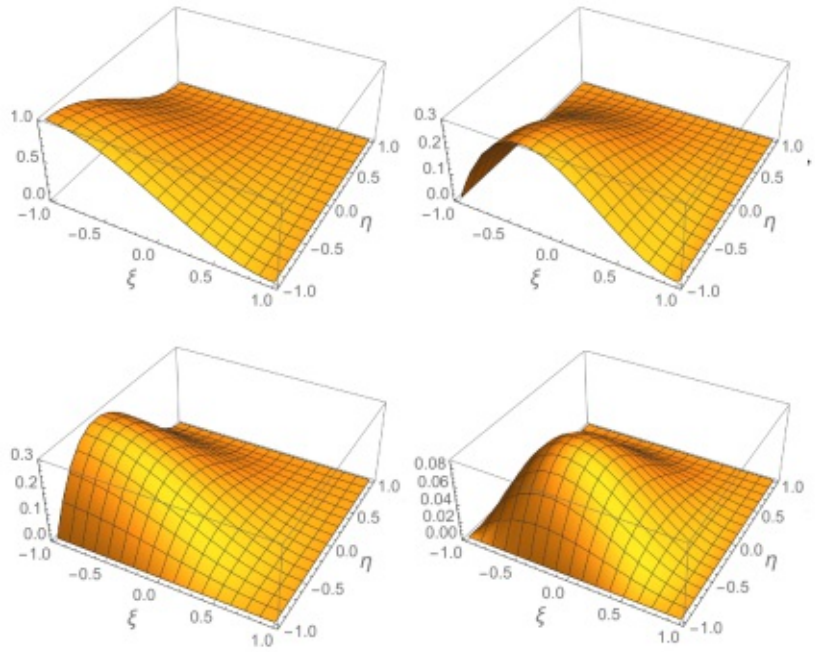


Fig. 2.4: The four Bogner-Fox-Schmit shape functions for node i, defined over the reference element with $\xi, \eta \in [-1, 1]$. Adapted from [19].

2.3.3 Linear Buckling Analysis in FEM

As mentioned in the introduction, a complete non-linear formulation of stability problems can be expensive in terms of computational resources. Under certain conditions, it is possible to obtain a good estimate of the critical load with a linear buckling analysis, which shall be introduced in this section, following [14]. It is especially suitable for treating stability problems of thin-walled structures like beams, plates and shells, because loss of stability can already occur as they undergo only small displacements. The method itself is valid also for continua and other structural elements like beams or shells. Here, it is applied to plates, as their stability behavior is of interest in this thesis.

Starting from a non-linear formulation of the principle of virtual work additional stiffness terms arise, that can be added to the linear stiffness matrix that was derived in Eqs. (2.20) and (2.35) to form the tangential stiffness matrix needed in non-linear analysis. A basic assumption for linear buckling analysis is that prior to the loss of stability only small displacements and small displacement gradients occur. This leads to the fact that all additional stiffness terms, that are of higher order in the displacement gradient, can be neglected, and only a single additional term must be taken into account. This term describes the effect of the present stress on the stiffness of the structure and is also called the geometrical contribution.

The bilinear form from plate theory as introduced in Eq. (2.29) is revisited and the stress stiffening is considered with the second term, written as

$$\mathcal{B}(w, \delta w) = \int_\Omega (\boldsymbol{\mathcal{L}} \delta w^{\mathrm{T}}) \mathbf{D} (\boldsymbol{\mathcal{L}} w) \, \mathrm{d}\Omega + \int_\Omega (\boldsymbol{\mathfrak{L}} \delta w^{\mathrm{T}}) \boldsymbol{\Sigma} (\boldsymbol{\mathfrak{L}} w) \, \mathrm{d}\Omega, \tag{2.39}$$

where $\mathscr{L}$ is another linear differential operator, representing the gradient, as in Eq. (2.40).

$$\mathscr{L} = \begin{bmatrix} \dfrac{\partial}{\partial x_1} \\ \dfrac{\partial}{\partial x_2} \end{bmatrix} \tag{2.40}$$

$\boldsymbol{\Sigma}$ is an extended matrix form of the Cauchy stress tensor $\boldsymbol{\sigma}$ and introduces the stress stiffening effect. Both terms from Eq. (2.39) can be used for the formulation of stiffness matrices, where the first one yields $\mathbf{K}_\mathrm{P}$ as in Eq. (2.35), and the second one yields $\mathbf{K}_\mathrm{G}$. In the wide body of literature different notions appear for $\mathbf{K}_\mathrm{G}$, among which are "geometrical stiffness matrix" and "initial stress stiffness matrix". In this thesis, the latter is used exclusively.

With similar arguments as in Sections 2.1 and 2.2, the elemental formulation follows from the bilinear form as

$$\boldsymbol{k}_\mathrm{G} = \int_{-1}^{1}\int_{-1}^{1} [\mathscr{L}\mathbf{N}^\mathrm{T}]\boldsymbol{\Sigma}_e[\mathscr{L}\mathbf{N}]\ \mathrm{det}\mathbf{J}\ \mathrm{d}\xi\mathrm{d}\eta. \tag{2.41}$$

Now, $\boldsymbol{\Sigma}_e$ is defined as the stress matrix that is obtained in the state of plane stress for a unit load. We further introduce a load factor λ, such that the stress matrix at the critical point may be written as $\boldsymbol{\Sigma}_c = \lambda\boldsymbol{\Sigma}_e$. In the critical state a bifurcation of the equilibrium path takes place, such that we may judge stability by seeking adjacent equilibria. Following the introduced notation, two possible equilibria can be denoted as

$$[\mathbf{K}_\mathrm{P} + \lambda\mathbf{K}_\mathrm{G}]\,\mathbf{U_1} = \mathbf{F}, \tag{2.42}$$

$$[\mathbf{K}_\mathrm{P} + \lambda\mathbf{K}_\mathrm{G}]\,\mathbf{U_2} = \mathbf{F}. \tag{2.43}$$

Subtraction of those two expressions yields

$$[\mathbf{K}_\mathrm{P} + \lambda\mathbf{K}_\mathrm{G}]\,(\mathbf{U_1} - \mathbf{U_2}) = \mathbf{0}, \tag{2.44}$$

which forms a generalized eigenvalue problem, since $\mathbf{U_1} - \mathbf{U_2} \neq \mathbf{0}$ by definition. Solving Eq. (2.44) results in n eigenvalues, which belong to the load levels at which buckling occurs. Usually, the lowest eigenvalue is of interest, as it describes the lowest load to cause loss of stability.

This generalized eigenvalue problem can be re-written as $\mathbf{K}_\mathrm{P}\boldsymbol{\Phi} = -\mathbf{K}_\mathrm{G}\boldsymbol{\Phi}\boldsymbol{\Lambda}$ to bring it closer to the usual notation used in MATLAB. $\boldsymbol{\Phi}$ describes the eigenvectors and $\boldsymbol{\Lambda}$ are the corresponding eigenvalues.

2.3.4 Error Sources

The displacement-based Finite Element Method approximates the displacement field. Stresses are obtained after computing strains, which result as derivatives from the already approximated displacements. Therefore, they may be even more inexact than the obtained displacement field, but different strategies exist to minimize the error. Generally speaking, these fall into the following two categories, that may be used separately or even in combination [20]:

1. *h-refinement*, in which the same element formulation is used, but the elements are changed in size. In some locations they are made smaller and, to increase overall economy, in others also bigger.

2. *p-refinement*, in which the same element size is used, but the order p of the polynomial in the element definition is increased, generally in a hierarchical way.

The final error can originate from different sources, most notably discretization error, integration error and round-off error. Discretization error is introduced as soon as the real, continuous problem with infinite degrees of freedom shall be approximated with only a finite set thereof. It arises when the true displacement field cannot be approximated within the problem's constraints, which are the number and type of elements used as well as boundary conditions and loading. Increasing the number of degrees of freedom, either by h- or p-refinement, we expect the solution to approximate the true one more accurately (subject to the condition that the true displacement can be represented) [13]. Integration error occurs due to the numerical integration needed to evaluate the system matrices introduced in Eqs. (2.20) and (2.35). It is usually limited in traditional FEM-applications but plays a paramount role in FCM due to the adaptive integration procedure needed, see Section 2.4. Round-off error originates in floating point arithmetic and is constantly present in all computer algebra software, but usually of very limited amount. Some applications allow to increase decimal places, for example from *single* to *double* representation, but other than that it can hardly be prevented.

Errors in the obtained results can only be determined when reference solutions are available. In the best case they exist in analytical form, but they may also be obtained from commercial FEM software, due to their reliability proven over decades. To do so, "overkill" analyses are performed, where an extremely fine discretization is used. The error committed then can be analyzed most clearly by using convergence diagrams, which visualize the error as a function of some parameters, like mesh size.

2.4 Finite Cell Method

A thorough introduction to the FCM can be found in many contributions, such as [10] or [21]. Here, only the fundamental ideas are outlined, based on [21]. Not only for the sake of simplicity, but also because of its importance for this thesis, two-dimensional linear elasticity is used as an introductory example. It is noted however, that FCM is also applicable when dealing with many other partial differential equations in engineering or physics.

In the FCM, a larger, *extended domain* Ω_e is introduced, into which the *physical domain* Ω_{phys} is embedded. Ω_e is usually chosen to have a simple and regular shape. This way, also the *fictitious domain* is introduced as $\Omega_{fict} = \Omega_e \setminus \Omega_{phys}$. These three different domains are depicted in Fig. 2.5. An indicator function $\alpha(\boldsymbol{x})$ is used to distinguish between the physical and the fictitious domain. $\alpha(\boldsymbol{x})$ is set to 1.0 for all $\boldsymbol{x}$ inside Ω_{phys} and to 0 for $\boldsymbol{x}$ inside Ω_{fict}:

$$\alpha(\boldsymbol{x}) = \begin{cases} \alpha_1 = 1.0, & \forall \boldsymbol{x} \in \Omega_{phys} \\ \alpha_0 = 0.0, & \forall \boldsymbol{x} \in \Omega_{fict}. \end{cases} \tag{2.45}$$

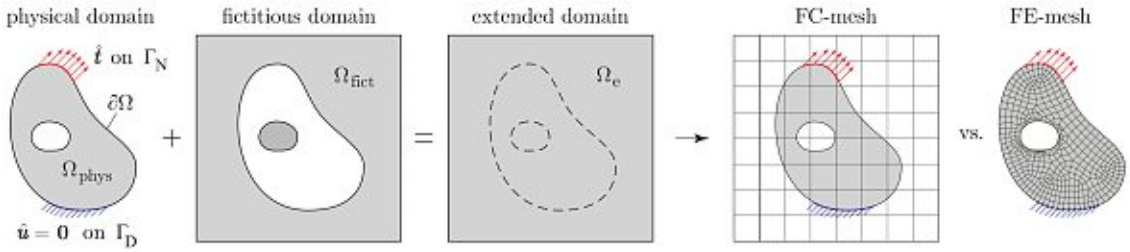


Fig. 2.5: Visualization of different domains and a typical discretization used in FCM on the left, a typical discretization for FEM on the right [21].

In practical applications, $\alpha_0 \ll 1$ is often used to prevent possible ill-conditioning of the resulting system matrices. In this thesis, $\alpha_0 = 1 \times 10^{-10}$ is used, as suggested in several publications [10, 21].

The original problem as introduced in Eq. (2.7) must now be computed for the complete extended domain. The fictitious domain, however, shall not have any effect on the behavior of the original problem. To ensure this, the indicator function $\alpha(\boldsymbol{x})$ is used to scale the stiffness (and body forces, if present) accordingly, meaning that material inside the fictitious domain gets substantially more compliant than material inside the physical domain:

$$\mathcal{B}(\boldsymbol{u}, \delta\boldsymbol{u}) = \int_\Omega [\mathbf{L}\delta\boldsymbol{u}]^\mathrm{T} \alpha \mathbf{C} [\mathbf{L}\boldsymbol{u}] \, \mathrm{d}\Omega, \tag{2.46}$$

$$\mathcal{F}(\delta\boldsymbol{u}) = \int_\Omega \delta\boldsymbol{u}^\mathrm{T} \alpha \boldsymbol{b} \, \mathrm{d}\Omega + \int_{\Gamma_\mathrm{N}} \delta\boldsymbol{u}^\mathrm{T} \boldsymbol{t}_\mathrm{N} \, \mathrm{d}\Gamma_\mathrm{N}. \tag{2.47}$$

One of the most cumbersome tasks when performing an FEM analysis often is the discretization procedure. Additionally, distorted elements in the mesh cause integration errors. While this task is greatly simplified and distorted elements do not occur in FCM, the main difficulty now lies in the computation of the system matrices and incorporation of boundary conditions. To distinguish between FEM and FCM all discretized entities are now denoted as *cells*. As can be seen in Fig. 2.6, all cells are classified into three types: physical cells, fictitious cells, and cut cells. Physical cells directly relate to finite elements, as they are located inside Ω_{phys} completely; related cell matrices are also computed exactly the same way. Fictitious cells lie inside the fictitious domain completely and the difference to finite elements is that their stiffness would be scaled with α_0. In practical application however, fictitious cells are deleted as soon as they are identified, because they no longer serve any purpose.

In all cut cells, there is a transition between α_0 and α_1, introducing discontinuous integrands when computing the system matrices. This demands adapted integration techniques. Gauss-Legrende quadrature, as typically used in FEM, serves very well for continuous integrands but introduces notable integration error for discontinuous integrands. In [21], Petö et al. review integration methods for discontinuous integrands and refer to the wide body of related literature. Numerical computation of discontinuous integrals is still actively researched and the main feature of [21] is the proposal of a new integration scheme based

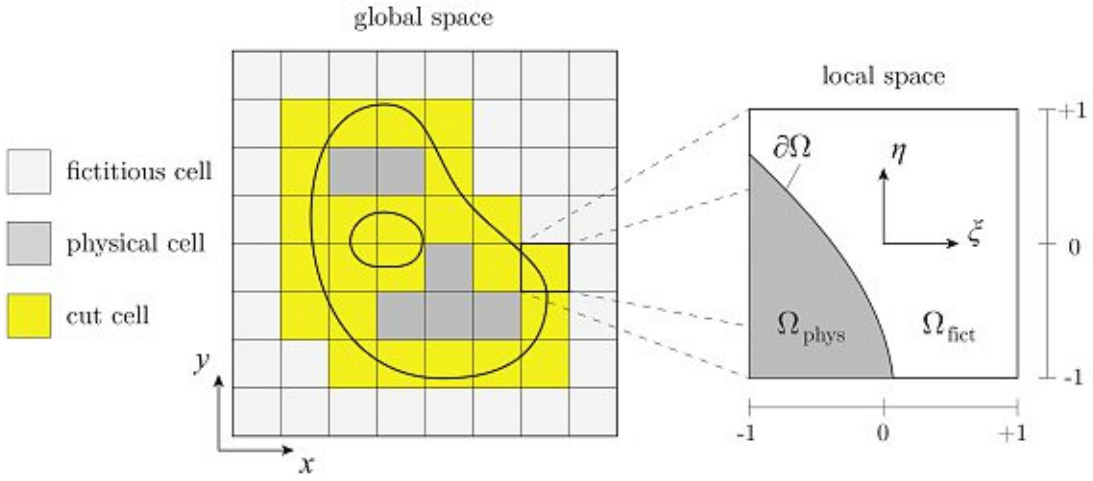


Fig. 2.6: Types of sub-cells on the left, a cut cell in local coordinates (ξ, η) on the right [21].

on image-compression techniques. It is based on the traditional adaptive integration scheme of introducing a local integration mesh, which is therefore also explained in great detail.

This thesis applies a traditional adaptive integration scheme via a *quadtree-decomposition*. In all cut cells a local integration mesh is introduced, which subsequently creates smaller and smaller sub-cells near the cut line (the discontinuity), see Fig. 2.7. Each cut cell is divided into four equal sized quadrants and this procedure is repeated until a previously defined *refinement level* k is reached. All sub-cells are again classified as physical, fictitious, and cut and their sub-cell matrices computed accordingly. The sum of the integrals over the integration sub-cells then yields the integral over the original cell, e.g., the stiffness matrix of a specific cell is obtained as sum over all n_{sc} sub-cell's stiffness matrices:

$$\mathbf{K}^c = \sum_{sc=1}^{n_{sc}} \mathbf{K}^{(c,sc)}. \tag{2.48}$$

$\mathbf{K}^{(c,sc)}$, the stiffness matrix of a specific sub-cell sc in a cell c is computed in local parametric coordinates r and s, where again $r, s \in [-1; 1]$. It is computed as in Eq. (2.20), but due to the usage of these local coordinates of the sub-cells, the integrand is extended by an additional term $\det(\mathbf{J}^{sc})$, describing the geometry mapping between cell and sub-cell coordinates. Also, the scaling parameter α must be included:

$$\mathbf{K}^{(c,sc)} = \int_{-1}^{1}\int_{-1}^{1} [\mathbf{L}\mathbf{N}^{\mathrm{T}}]\alpha\mathbf{C}[\mathbf{L}\mathbf{N}] \det(\mathbf{J}^{sc}) \det(\mathbf{J}) \, dr ds. \tag{2.49}$$

The integral in Eq. (2.49), as well as all other area integrals in this thesis, is evaluated using numerical integration and 4x4 Gauss-Lobatto integration points. The corresponding Gauss points are located at $x_i = [-0.861136, -0.339981, 0.339981, 0.861136]$ for the one-dimensional integration; the integration weights are $w_i = 0.652145$ for $x_i = \pm 0.339981$ and $w_i = 0.347855$ for $x_i = \pm 0.86113$. Gauss nodes and weights for the necessary two-dimensional integration are obtained by simple matrix manipulation of these data.

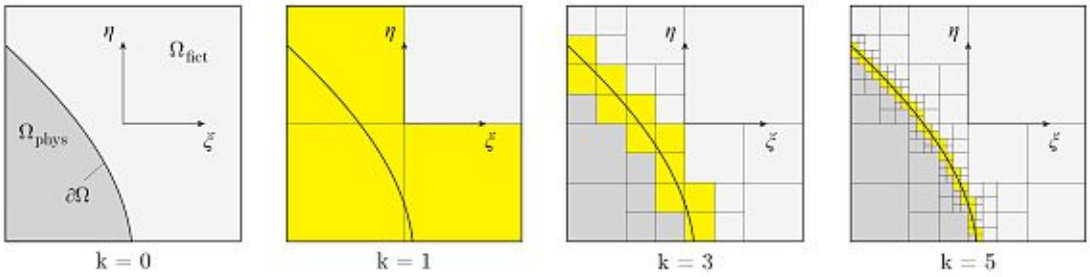


Fig. 2.7: Quadtree decomposition of a cut cell, refinement level $k = 0$ to $k = 5$ [21].

It shall be noted that the local integration mesh and the refinement process are introduced for the purpose of numerical integration only. The total number of DOF in the system is not increased, as the refinement does not affect the approximation of the primary field variables, which is still defined by the nodal degrees of freedom of the original mesh.

The cut line, as depicted in Fig. 2.6, can be defined using different approaches. The first one, also used in thesis for circular cutouts, describes the geometry implicitly. For a circular area, this means that the inequality $(x - x_c)^2 + (y - y_c)^2 < r^2$ is implemented in a level set method function. This function returns the information, if a probed point is inside or outside the so defined circle and is used to categorize all cells into the three introduced categories.

The second approach to describe the cut line is point-wise, where the geometry between the points is interpolated using linear, quadratic or higher order line segments. This of course allows much greater flexibility in defining the cutouts, as can be seen in Fig. 2.8, where a shark shape is cut into a square structure. Said method can also be used to describe the geometry from image files or voxelized data. Possible applications are, among others, the analysis of porous foam structures or bio-medical problems from scan data.

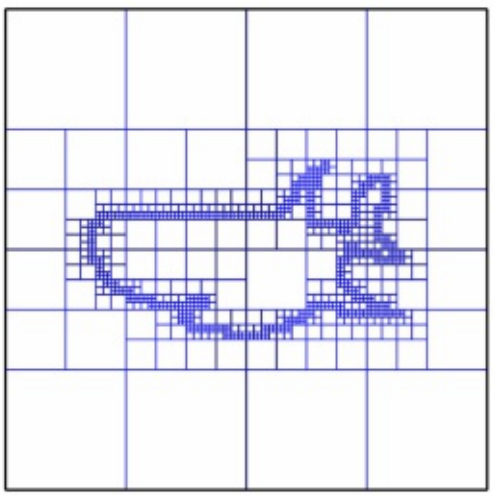


Fig. 2.8: Shark-shaped cutout, defined by polygon cut lines, and its quadtree-based refinement ($k = 6$) [22].

Chapter 3

Implementation

The implementations are based on a scientific MATLAB code provided by Dr. Eisenträger from the University of New South Wales [22]. Its main feature is the analysis of plate bending problems under different loading conditions using BFS-elements, both with and without FCM. Code without FCM of course is a traditional FEM code in this sense. The main steps towards extending the provided code for optimizing stability boundaries of plates, which were performed in the framework of the present master's thesis, are as follows:

1. The implementation of solving the plane stress problem, including the computation of strains and stresses from the displacements. Necessary information from this step is the stress field.
2. Verification and validation of the results from step 1 by comparison with analytical solutions, examples from the literature and commercial software.
3. The computation of the initial stress stiffness matrix $\mathbf{K}_{\mathrm{G}}$ for the plate problem, which requires the stress field from step 1.
4. Solving the generalized eigenvalue problem with the plate stiffness matrix $\mathbf{K}_{\mathrm{P}}$ and the plate initial stress stiffness matrix $\mathbf{K}_{\mathrm{G}}$. The resulting smallest positive eigenvalue is the critical load factor for buckling. The related eigenvector describes the corresponding mode shape.
5. Verification and validation of the results from step 4 by comparison with analytical solutions, examples from the literature and commercial software.

As soon as the critical buckling loads can be computed with satisfactory precision, analyses with varying positions for the cutout are performed in Chapter 4. Varying the position all over the plate geometry on pre-defined grid points results in a function of buckling loads. It can be used to assess the effect of the introduction of the hole. Positions that increase the buckling strength become apparent, especially after the application of proper visualization techniques. Resulting plots will be used for comparison with the sensor-hole concept from Gracia and Rammerstorfer [7].

MATLAB (abbrv. for "matrix laboratory") is a commercial software suite, including a programming language and numeric computing capabilities. It is used in this thesis almost exclusively, making use of its straight-forward matrix manipulating and plotting functionalities. WOLFRAM MATHEMATICA, another commercial software suite, is used for verification of simple FEM problems, as well as for the visualization of shape functions and the plate problems depicted in Figs. 2.4 and 3.5.

Table 3.1: Numbering of nodal degrees of freedom for the state of plain stress.

1	2	3	4	5	6	7	8
u	v	$\frac{\partial u}{\partial x}$	$\frac{\partial v}{\partial x}$	$\frac{\partial u}{\partial y}$	$\frac{\partial v}{\partial y}$	$\frac{\partial^2 u}{\partial x \partial y}$	$\frac{\partial^2 v}{\partial x \partial y}$

3.1 State of Plane Stress

The first extension of the provided code is the implementation of plane stress analysis. The primary variables are the in-plane displacements u and v. Most commonly in FEM, they are approximated by nodal values of u and v only, using linear or quadratic shape functions [15, 23]. Pursuing a holistic concept in this thesis, BFS shape functions are also applied for the state of plane stress. To do so, additional DOF are introduced, which increases computational cost. Beneficial, however, is the improved convergence behavior and the seamless link to the plate problem in terms of shape functions. These additional DOF are the displacements' first derivatives and mixed second order derivatives in x- and y-direction, see Table 3.1. This results in eight DOF per node or 32 DOF per element.

In the first step, a pure FEM implementation of BFS elements is conducted. Achieving this goal required adapting multiple parameters and functions in the existing MATLAB code, but other than that the structure and functionalities of the provided code are maintained. Most challenging is the correct definition of new routines to evaluate the shape functions, now for the two variables u and v. Both are interpolated exactly as w is in the case of the plate problem, and this similarity is also reflected in the code. Additionally, it must be accounted for a new type of loading conditions, which are in-plane line loads acting on single or multiple lines. The code must ensure that computed nodal loads are energetically equivalent to these line loads, complying with Eq. (2.9).

For verification of the results, analytical solutions are available for uniaxial tension or compression and beam theory offers suitable comparative results for long and slender structures. A beam-like problem is also compared with an alternative FEM solution implemented using WOLFRAM MATHEMATICA and yields identical results. For the sake of brevity we do not present these results here, but rather proceeded with the validation of our implementation of FCM in combination with plane stress analysis.

3.1.1 Problem Definition

A well known problem for benchmarking plane stress analysis in context of FCM is depicted in Fig. 3.1 [10, 21]. A perforated square structure is statically loaded at its top edge and constrained with Dirichlet boundary conditions at its bottom and left edge. A linear-elastic and isotropic material behavior is assumed. The Young's modulus is $E = 206.9$ GPa and the Poisson's ratio is $\nu = 0.29$. Parvizian et al. [10] report an highly exact estimate for the strain energy of $\mathcal{S}_{\text{ref}} = 0.7021812127$, which was achieved using p-FEM and the blended function method for exact geometry representation.

p-FEM is an extension of traditional FEM, where a fixed mesh is used and the polynomial degree p is allowed to increase. It can be shown that p-extension on properly designed meshes can realize exponential rates of convergence [24], which allows the highly exact reference measure in this case.

The comparison to the reference strain energy is used for a global investigation of this implementation's results. The relative error e for the strain energy $\mathcal{S}$ is computed as $e = |\mathcal{S} - \mathcal{S}_{\text{ref}}|/\mathcal{S}_{\text{ref}}$. Likewise, the relative error compared to a generic reference solution a_{ref} for all following analyses is computed as $e = |a - a_{\text{ref}}|/a_{\text{ref}}$. Additionally, a *von Mises* stress plot is created for visual verification.

During the implementation process a fallback to a more simple adaptive integration method was performed. With this method, not every single integration point is checked whether it is inside or outside the physical domain. Instead, an indicator value of $\alpha = 1$ is assigned to *all* cut cells, which come from the same quadtree decomposition. In a nutshell, this way all cells and sub-cells, that simply "touch" the physical domain are used for computation; only cells and sub-cells completely inside the fictitious domain are disregarded. This concept is commonly referred to as "Voxel-FEM" or "Voxel-FCM" , as each voxel is a finite element/cell; the geometry is described in a staggered way. Voxel-FEM is successfully applied, among others, to the analysis of seismic ground motion in [25] and Voxel-FCM is studied in [26]. The fallback performed here aids in a better understanding of the method and allows direct traceability of the results. The number of refinements needed for acceptable accuracy however is increased drastically, as well as computational power needed. For that reason only discretizations with a small number of elements are used, which is perfectly fine for this demonstration purpose. It is now also possible to draw a direct comparison with an in-house FCM-MATHEMATICA solution, which serves as a first check of the FCM-MATLAB code.

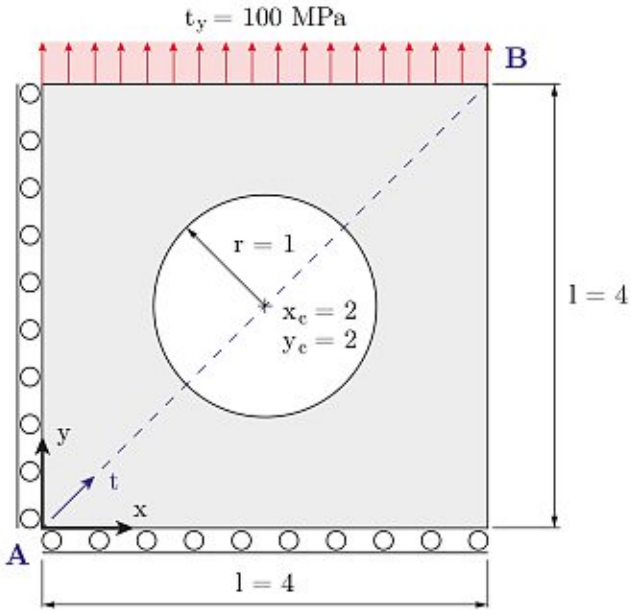


Fig. 3.1: Reference problem for the state of plane stress. Perforated plate under static loading, with dimensions given in [mm] [21].

As can be seen in Fig. 3.3, the results from the MATLAB code asymptotically converge to the MATHEMATICA results for the strain energy. A noteworthy aspect in Fig. 3.3 is the impact of the discretization error. With $n = n_x^2 = 4$ elements only, a relative error of $e \approx 6 \times 10^{-2}$ cannot be exceeded, regardless the number of refinements, which reduces integration error only. The numerical results are also shown in Table 3.2.

3.1.2 Results

With satisfying results from the comparison in Fig. 3.3, the switch back to the original integration method, as introduced in Section 2.4, is now performed and the relative error is again compared to the the benchmark solution. The results for four different discretizations are plotted in Fig. 3.4. It can immediately be seen, that the asymptotic convergence behavior previously seen in Fig. 3.3 is lost, which is typical for this type of adaptive integration scheme in combination with FCM. Additionally, a significant reduction of needed refinements k to achieve the same level of accuracy is noted.

If we consider the discretization with $n_x = 4$, the converged solution with the Voxel-FCM shows an error of $e \approx 5 \times 10^{-3}$ after more than ten refinements steps (blue line in Fig. 3.3). The error when using the adaptive integration scheme equals $e_{AIS} \approx 5 \times 10^{-3}$ right from the first steps (light gray line in Fig. 3.4, Table 3.3) on, displaying its power. The steady horizontal line for the following refinement steps indicates, that the integration error is already minimized and that the discretization error cannot be overcome.

The von Mises equivalent stress is a stress measure usually used to predict yielding of metallic materials. It can be plotted on the contour of the structure and used as an additional measure of comparison. In Fig. 3.2 the perforated plate is displayed in its deformed configuration, including the von Mises plot. Shown displacements as well as the stress distribution match the expectations as well as comparative ABAQUS simulations, allowing to advance to the implementation of the buckling problem in the next section.

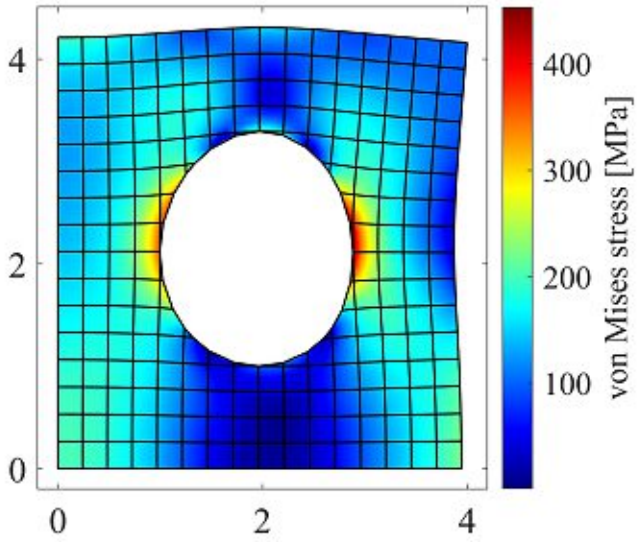


Fig. 3.2: Von Mises stress plot for the reference problem.

Table 3.2: Numerical results for the reference plane stress problem, obtained with the Voxel-FCM. Relative error e in strain energy.

Mesh	Level of Refinement k					
n_x	1	2	3	4	5	6
2	4.4934e-01	3.8098e-01	2.5778e-01	1.6843e-01	1.1551e-01	8.8224e-02
3	2.5125e-01	2.5125e-01	1.4419e-01	9.2879e-02	5.4042e-02	3.5410e-02
4	3.5081e-01	2.1347e-01	1.2060e-01	6.2397e-02	3.2414e-02	1.8071e-02
5	1.9196e-01	1.5031e-01	8.2847e-02	4.3473e-02	2.2724e-02	1.2637e-02
n_x	7	8	9	10	11	12
2	7.5360e-02	6.9183e-02	6.5868e-02	6.4384e-02	6.3620e-02	6.3242e-02
3	2.6542e-02	2.2107e-02	2.0065e-02	1.9047e-02	1.8550e-02	1.8303e-02
4	1.1343e-02	7.7922e-03	6.2185e-03	5.4100e-03	5.0123e-03	4.8211e-03
5	7.0312e-03	4.2694e-03	3.0042e-03	2.3745e-03	2.0618e-03	1.9112e-03

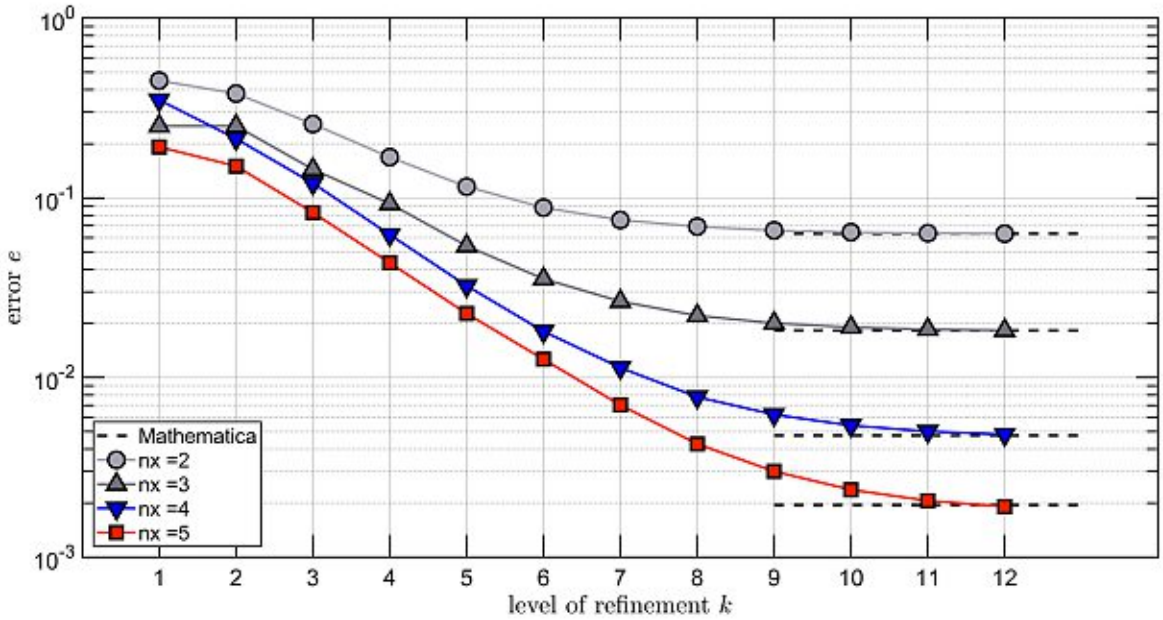


Fig. 3.3: Convergence of the Voxel-FCM solutions, including converged MATHEMATICA results.

Table 3.3: Numerical results for the reference plane stress problem, obtained with the advanced adaptive integration scheme. Relative error e in strain energy.

Mesh	Level of Refinement k				
n_x	1	2	3	4	5
4	2.7599e-03	5.3201e-03	5.5441e-03	4.7543e-03	4.6929e-03
8	5.0266e-06	1.1041e-03	2.1574e-04	2.1650e-04	7.2883e-05
16	1.7951e-03	2.2758e-06	7.9512e-05	6.6680e-05	4.6368e-06
32	9.0468e-04	2.5900e-05	7.6937e-05	8.9680e-06	1.0812e-05
n_x	6	7	8	9	10
4	4.5509e-03	4.6198e-03	4.6176e-03	4.6281e-03	4.6268e-03
8	1.3721e-04	1.3630e-04	1.4609e-04	1.4466e-04	1.4668e-04
16	6.7782e-06	3.0485e-06	1.5715e-06	3.6159e-06	4.0698e-06
32	9.5191e-07	2.4517e-06	4.0513e-07	4.8819e-08	5.1327e-08

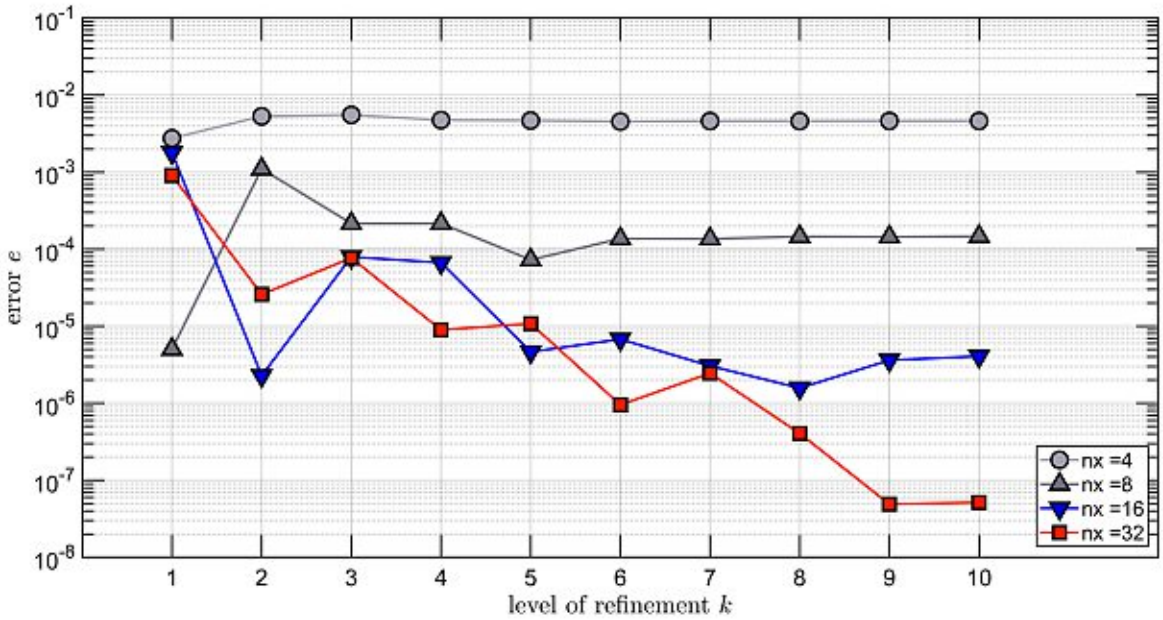


Fig. 3.4: Convergence of the advanced adaptive integration scheme solution.

3.2 Stability Behavior of Plates

The provided code already consists of a reliable implementation of the plate problem, both with and without FCM. This section describes the extension needed to examine the stability behavior of plates with linear eigenvalue buckling analysis. As introduced in Section 2.3.3, to do so a generalized eigenvalue problem with two stiffness matrices needs to be solved. The computation of the first one, the plate bending stiffness matrix $\mathbf{K}_\mathrm{P}$, is readily implemented. For the second one, the initial stress stiffness matrix $\mathbf{K}_\mathrm{G}$, additional steps must be undertaken.

$\mathbf{K}_\mathrm{G}$ is a term that originates from the nonlinear formulation of the principle of virtual work. It takes into account the effect of stress stiffening, meaning that it describes how a present stress field impacts the stiffness of the structure. The initial stress stiffness matrix can also be added, among other contributing terms, to the usual stiffness matrix, forming the tangential stiffness matrix for nonlinear analysis. It is available not only for plate problems, but a general feature of finite element analysis, or in an even broader sense, continuum mechanics [14].

Considering the key aspect of the adaptive integration scheme used here, it is important to note that the number of integration points rises rapidly with an increasing level of refinement. Also their location in space is not fixed, but object to the user-defined refinement. This is an important difference to traditional FEM, where the number and location of integration points is defined with mesh creation and does not change afterwards. In FEM, the stress may directly be computed after solving the plane stress problem and used afterwards for the determination of $\mathbf{K}_\mathrm{G}$. Here, it is considered more functional to move stress computation directly into the integration routine for the initial stress stiffness matrix, as all integration points are fully defined at this point. This approach for example allows different levels of refinements for the plane stress and plate problem respectively, which may be identified as a more computationally efficient procedure.

In Program Code 3.1 this MATLAB implementation is presented. In a `for`-loop over all integration points (IP) from line 6 on, the corresponding IP coordinates are computed from quadtree data. Then the computation of the shape function matrices takes place. In line 20 and 21 `Bg` is built from its constituent `Bl` by transformation, which is necessary as `Bl` is computed in local coordinates. `Bg` directly corresponds to the term $[\mathfrak{L}\mathbf{N}]$ in Eq. (2.41) from plate theory, meaning that the operator $[\mathfrak{L}]$ (the gradient) is applied on the BFS shape functions.
From line 23 to line 41 the stress computation is performed. The displacements within an element `UElem` from the plane stress problem are passed along to the integration routine to obtain strains. Therefore, `Bg_ESZ`[1] is built from plane stress shape functions, corresponding to the term $[\mathbf{LN}]$ in Eq. (2.20) from linear elasticity. This represents the standard strain-displacement operator applied on the shape functions. Strain (`eps`) is then obtained by matrix multiplication of `Bg_ESZ` and `UElem`, and stress (`sigma`) follows from matrix multiplication of the elasticity matrix `C` and `eps`. Finally, in line 55, `Bg` and `sigma` are used for the numerical integration, yielding $\mathbf{K}_\mathrm{G}$.

[1] the common German abbreviation “ESZ - Ebener Spannungszustand” for the state of plane stress is used for distinguishability

Program Code 3.1: Computatin of the *initial stress stiffness* matrix `Kg`

```
% Loop over subcells
for j = 1:nSC
    % Mapping from subcell to reference domain
    ...                                                % omitted for clarity
    % Loop over all integration points
    for i = 1:nIP
        % Compute the local coordinate of the integration point
        xi  = xiC  + 0.5*xiG(i)*hxi;
        eta = etaC + 0.5*etaG(i)*heta;
        % Jacobian matrix -- J = [dx/dxi, dy/dxi; dx/deta, dy/deta]
        J = JacobianMatrix(xi,eta,xElem,yElem);
        detJ = det(J);
        % Inverse of the Jacobian matrix
        Jinv = [J(2,2), -J(1,2); -J(2,1), J(1,1)]/det(J);
        % Matrix of the first order derivatives of the shape functions
        % w.r.t. the local coordinates
        Bl = [ Plate_SF_BFS_dxi(xi,eta)'   ; Plate_SF_BFS_deta(xi,eta)'] ;
        % Matrix of the first order derivatives of the shape functions
        % w.r.t. the global coordinates
        Bg = [ Bl(1,:)*Jinv(1,1) + Bl(2,:)*Jinv(1,2);
               Bl(1,:)*Jinv(2,1) + Bl(2,:)*Jinv(2,2) ];

        %% Compute stress at (xi,eta) from displacements
        % Matrix of the first order derivatives of the
        % shape functions for plane stress w.r.t. the local coordinates
        B1_ESZ = SF_BFS_dxi(xi,eta)'  ;
        B2_ESZ = SF_BFS_deta(xi,eta)' ;
        % Matrix of the first order derivatives of the shape
        % functions w.r.t. the global coordinates
        Bg1_ESZ = B1_ESZ*Jinv(1,1) + B2_ESZ*Jinv(1,2);
        Bg2_ESZ = B1_ESZ*Jinv(2,1) + B2_ESZ*Jinv(2,2);

        % Build third line by alternating entries & build complete Bg-matrix
        Bg3_ESZ = zeros(length(B1_ESZ),1)';
        Bg3_ESZ(1:2:end) = Bg2_ESZ(2:2:end);
        Bg3_ESZ(2:2:end) = Bg1_ESZ(1:2:end);
        Bg_ESZ = [ Bg1_ESZ; Bg2_ESZ; Bg3_ESZ ];

        % Compute strain and stress
        eps = Bg_ESZ * UElem;
        sigma = C * eps;

        % Global coordinates of the integration points
        [xIP,yIP] = Local2GlobalCoordinatesQuad(xi,eta,xElem,yElem);
        % Is the integration point in the fictitious or physical domain
        [FCM,Idx,Hole] = PointInMembershipTest(xIP,yIP,flag_MD,points);
        % Penalty-factor
        if FCM == 1 && Hole(Idx) == 1
            val = alpha;
        elseif FCM == 0
            val = 1;
        end

        % Stiffness matrix
        Kg = Kg + val*Bg'*sigma*Bg*h*detJ*detJsc*wG(i);
    end
end
```

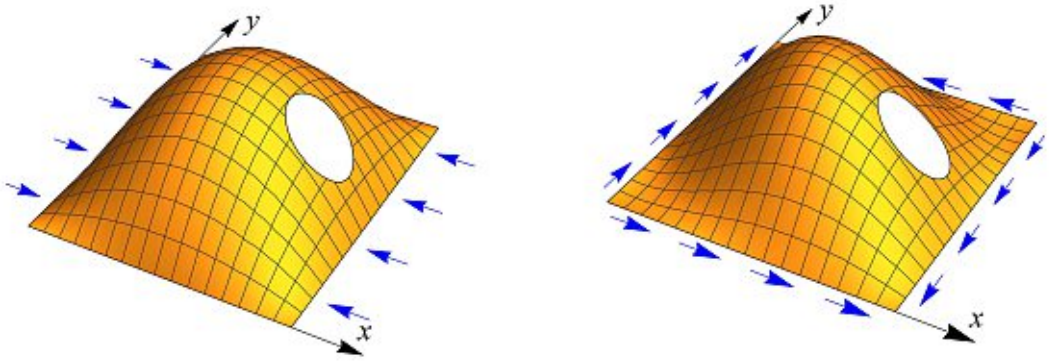
For solving the generalized eigenvalue problem $\mathbf{K}_\mathrm{P}\mathbf{\Phi} = -\mathbf{K}_\mathrm{G}\mathbf{\Phi}\mathbf{\Lambda}$ the MATLAB function `eigs` with default settings is used. `eigs` is a sophisticated routine to iteratively solve for a sub-set of eigenvalues and eigenvectors using a Krylov-Schur algorithm[2]. At this point it is also worth mentioning how MATLAB solves systems of linear equations of the type $\mathbf{A}\boldsymbol{x} = \mathbf{B}$ for $\boldsymbol{x}$. In this thesis, it is needed to solve $\mathbf{KU} = \mathbf{F}$ for the displacements $\mathbf{U}$. The function `mldivide`, simply executed as "\", is a versatile tool to do so. It analyzes the input matrices and dispatches the problem to an appropriate solver, aiming to minimize computation time[3].

3.2.1 Problem Definition

After ensuring the correct computation of the eigenvalues and eigenvectors for arbitrarily loaded, unperforated plates, the next step is to analyze plates with holes. Accordingly, an example problem is introduced here. A quadratic plate with the properties shown in Table 3.4 is studied, where a and b denote edge length and h is the plate thickness. Linear-elastic, isotropic material behavior is assumed again, where E is the elastic modulus and ν the Poisson's ratio. The circular cutout is parameterized with three parameters: the pair of coordinates (x_c, y_c), defining the center of the circle, and the radius r.

Table 3.4: Properties of the plate problem.

Material Properties		Geometric Properties			Hole Properties		
E [GPa]	ν [1]	a [m]	b [m]	h [m]	x_c [m]	y_c [m]	r [m]
210	0.3	1	1	1×10^{-3}	0.9	0.5	0.055



(a) Load case 1: uniaxial compression **(b)** Load case 2: pure shear

Fig. 3.5: A quadratic plate subjected to different in-plane line loads. Illustration by Yury Vetyukov.

[2]https://www.mathworks.com/help/matlab/ref/eigs.html
[3]https://www.mathworks.com/help/matlab/ref/mldivide.html

The plate is loaded with in-plane edge unit forces per unit length in two different load cases (LC). The first one is the load case of pure compression, where normal tractions act on two opposite edges, see Fig. 3.5a. The second one is pure shear loading, where shear tractions act on all four edges, see Fig. 3.5b. To avoid rigid body motion in the state of plane stress, a single node's u and v displacements are restricted. Other than that, no boundary conditions are needed due to the symmetry of the loads in both scenarios. For the plate problem, the displacement w is restricted at all boundaries.

3.2.2 Results

The reference solution is obtained by solving the same problems with a very fine mesh in ABAQUS, which outputs the eigenvalues with six significant digits. In ABAQUS the plate is discretized with 44 435 S8R5 elements, which are 8-node thin shell elements, using reduced integration, five degrees of freedom per node and quadratic interpolation. The resulting critical loads from these two analyses are 764.98 for LC1, uniaxial compression, and 1782.76 for LC2, pure shear. This information is also summarized in Table 3.5.

The FCM results for both defined load cases can be seen in Figs. 3.6 and 3.7 and Tables 3.6 and 3.7, where again the error is displayed as a function of discretization and refinement. While hardly any effect of the refinement level k can be noticed, regular convergence behavior is found regarding discretization. The quality of the FCM results is very satisfactory, the discretization with $n_x = 50$ elements yields a relative error of only $e \approx 1 \times 10^{-4}$ in the case of uniaxial compression, approaching levels of accuracy where even the precision of the 6-digit ABAQUS solution must be questioned.

This problem also serves to identify a suitable baseline analysis configuration in terms of mesh size and level of refinement. In all subsequent optimization runs, if not stated explicitly different, 16 elements along the structure's edges are used and five steps of refinement are performed. This is a combination that proved itself as satisfactory compromise between computational cost and accuracy here.

It is yet an unanswered question why the refinement level k appears to have so little effect on the solution of the stability problem. In the presented problems the size of the cutout is comparatively small. A possible explanation for this behavior may be that the disturbance originating from the holes can be accurately accounted for with little refinement. It should however be examined in more detail in future work.

Table 3.5: Setup and results for the ABAQUS reference analysis

Element		Critical Load Multiplier	
Type	Number	Load Case 1	Load Case 2
S8R5	44 435	764.98	1782.76

Table 3.6: Relative error e in the critical buckling load for LC1: uniaxial compression. Different mesh size and levels of refinement.

Mesh	Level of Refinement k				
n_x	1	2	3	4	5
8	4.2116e-03	3.9969e-03	4.1425e-03	4.1967e-03	4.1948e-03
16	1.2057e-04	5.1038e-04	7.0205e-04	6.4900e-04	6.5554e-04
32	1.8031e-04	2.9684e-04	2.3599e-04	2.4807e-04	2.5422e-04
50	3.9714e-04	1.2387e-04	9.1792e-05	8.4381e-05	7.9527e-05

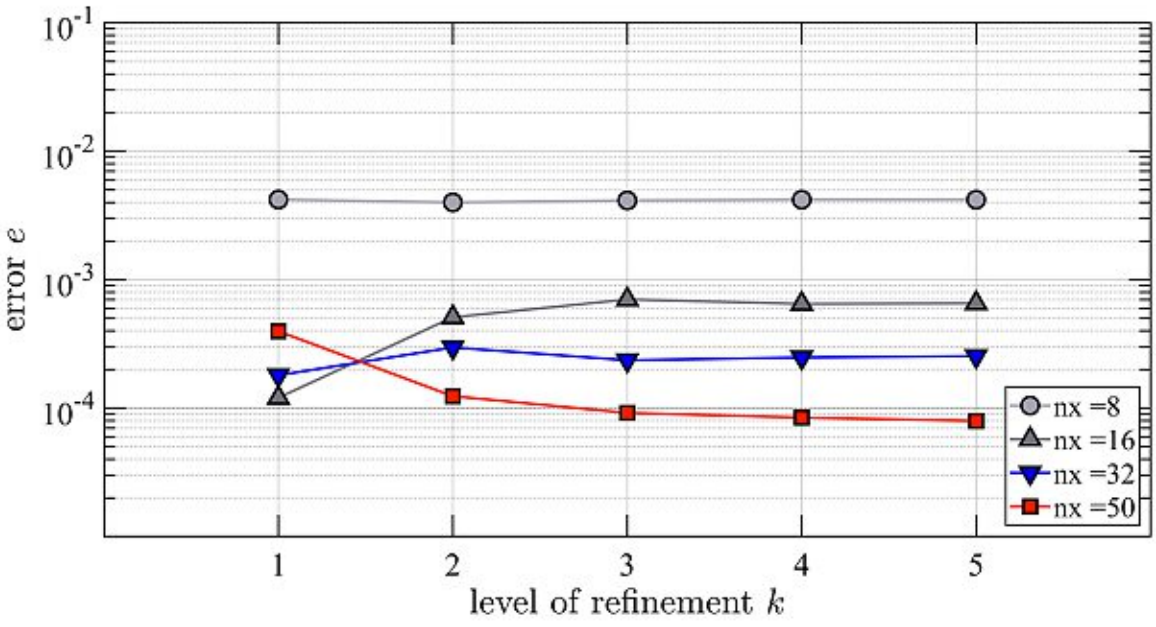


Fig. 3.6: Convergence of the critical buckling load for LC1: uniaxial compression.

Table 3.7: Relative error e in the critical buckling load for LC2: pure shear. Different mesh size and levels of refinement.

Mesh	Level of Refinement k				
n_x	1	2	3	4	5
8	4.7532e-02	4.8014e-02	4.7997e-02	4.7967e-02	4.7970e-02
16	1.2171e-02	1.2337e-02	1.2306e-02	1.2333e-02	1.2333e-02
32	2.8378e-03	2.7893e-03	2.8401e-03	2.8375e-03	2.8352e-03
50	9.2939e-04	9.7370e-04	9.6785e-04	9.7303e-04	9.6903e-04

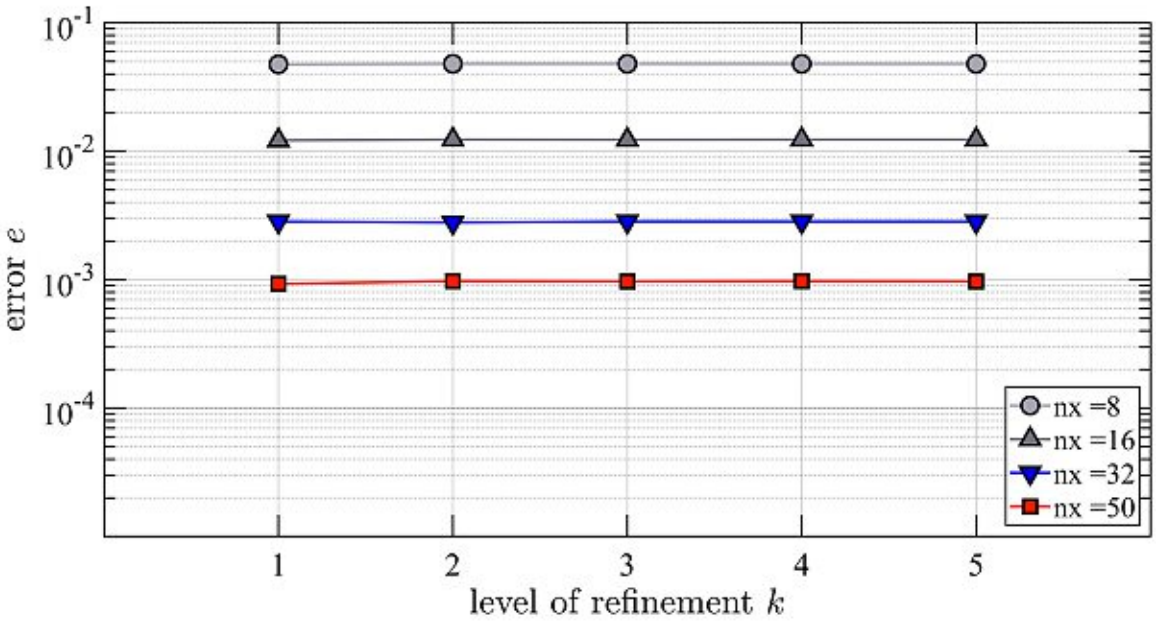


Fig. 3.7: Convergence of the critical buckling load for LC2: pure shear.

Chapter 4

Optimization

This chapter describes the application of the so far introduced methodology for optimizing the stability boundaries of plates with cutouts. Under optimization we understand that the most beneficial position of a hole in regards to the buckling strength of a plate is sought. Different methods can be used for this search, among which *optimization algorithms* immediately come to mind. A comprehensive monograph on optimization methods in general is [27], and their numerical implementation can be found most clearly in [28]. Such algorithms may very well solve the optimization problem at hand and converge to the solution of the *best* position. Disadvantageous, however, is that many efficient optimization methods require the computation of the objective function's gradient, which is not feasible in this case. On the other hand, derivative-free algorithms are, generally speaking, costly. The Nelder-Mead simplex algorithm [29] is such a derivative-free method, it's working principle is a simplex of n points, where at all of which the function is evaluated. The current worst point is discarded and new points are accepted into the simplex, until the convergence criteria are fulfilled.

This thesis however intentionally follows a different approach, where not only the best position is found, but *all* admissible positions are evaluated. This way, also the most disadvantageous position is identified as well as the buckling load as a function of the position of the hole, defined over the plate geometry. This function is perfectly suitable for plotting, thus allowing a quick and reliable assessment of how a hole affects the stability behavior of such plates with cutouts.

4.1 Optimization Setup

The main idea is that the stability problem introduced in Section 3.2 is solved repeatedly for different hole positions. Present thesis is restricted to circular cutouts with a fixed radius of $r = 0.055 \cdot a = 0.055$ m. This is considered small enough to enable a direct comparison with the sensor-hole concept presented in [7], as well as big enough to considerably affect the critical load. Cutouts with different geometries and size are open to future investigations, for which an adaption of the mesh size and refinement should be considered to find an adjusted parameter set that allows efficient analysis.

For the optimization carried out here, the properties stated in Table 3.4 remain valid, except for the hole position parameters x_c and y_c. To evaluate the critical load at all admissible hole positions, at first the region is defined, in which introducing a cutout is allowed. Here, this region is defined inside the square plate with edge length $a = 1$ m as $\mathcal{A} = [0.1, 0.9] \times [0.1, 0.9]$, such that a minimal distance to the edges is ensured. The region $\mathcal{A}$ inside the plate can be seen in Fig. 4.1. It follows that the complete surface area

of the circular cutout is inside the plate and does not interfere with the topology of the edges. This is important on account of the numerical load scheme, that is not able to handle disconnected edges in this implementation. Additionally, the remaining member width in the vicinity of the cutout is assumed big enough such that local loss of stability is excluded from this analysis.

The region $\mathcal{A}$ is then discretized by covering the geometry with an equally-spaced point grid with a fixed stepsize δ. The point grid defines the cutout positions and all point coordinates are successively handled to the main function for evaluation of the critical load. All point coordinates and associated buckling loads are then stored in a MATLAB-double in matrix form, readily available for visualization. After some initial attempts with different stepsizes, $\delta = 0.0125$ demonstrated sufficient smoothness of the resulting critical loads with reasonable computational effort. It is therefore chosen for the following optimization, completing the set of discretization parameters for this problem as also shown in Table 4.1. It should be recalled that n_x denotes how many elements are used along a single edge of the extended domain (coinciding with the outer plate dimensions). k describes how many steps of refinement are carried out in the FCM integration routine.

MATLAB offers a very useful data type for an efficient implementation of the repeated analyses. The code used in Section 3.2 is reformulated in a MATLAB-function that returns the critical load after handling the cutout position variables as input parameters. This function is then used in a so-called function handle[4], which passes a function to another function. It can elegantly be used for the evaluation of mathematical expressions over a range of values, which is exactly what is utilized here.

After the first few optimization runs it was quickly identified that both load cases show a certain symmetry for the critical load. Considering LC1 in Fig. 3.5a it seems clear, that a cutout at the position $[0.1, 0.5]$ and $[0.9, 0.5]$ must have the same effect on the buckling strength. Identically, a cutout at $[0.1, 0.1]$ and $[0.1, 0.9]$ leads to the same behavior, and the problem shows double symmetry in respect to a vertical and horizontal bisecting line going through the plate's midpoint. The same is true for LC2, where the symmetry lines are the two diagonal lines that connect two opposite corner points each. It is noted that present implementation is capable of capturing these symmetries.

For improving the overall efficiency, only the appropriate quarter of the plate is discretized with the mentioned point grid and the needed number of analyses is reduced by a factor of 4. This can again be seen in Fig. 4.1, where the quarter plate is discretized with a point grid of stepsize $\delta = 0.025$ (chosen for visualization) for both LC1 and LC2. The remaining results needed for visualization are obtained by twofold mirroring of the original data.

Table 4.1: Discretization parameters.

Mesh n_x	Level of Refinement k	Stepsize δ
16	5	0.0125

[4] https://www.mathworks.com/help/matlab/function-handles.html

4.2 Post-Processing & Visualization

It is assumed that the critical buckling loads in terms of the cutout positions origin from a smooth functional. The obtained results however show small oscillations which are deemed as an unphysical effect, rather originating from discretization. The application of a filter is therefore considered admissible, and in fact, a Savitzky-Golay filter is effective to reduce this noise. The principle of Savitzky-Golay filtering is the "replacement of each point of a signal by some combination of the signal values contained in a moving window centered at the point, on the assumption that nearby points measure nearly the same underlying value"[5]. The replacement is based on fitting a n-th order polynomial through m points inside the window by the method of least-squares. The central point of the fitted polynomial then serves as the new smoothed data point.

A polynomial order of $n = 3$ through $m = 13$ points results in satisfactory filtering for present application. It should be noted, that if the input data is in matrix form, as in this thesis, the Savitzky-Golay filter operates on each column only. The design of a filter operating also on each row is conceivable, however not considered necessary here.

Different options exist for a useful visualization of the obtained results. The critical buckling load can be directly plotted over the x- and y-coordinates of the cutout origin to create a 3-dimensional plot, as can be seen in Fig. 4.2 for LC2: pure shear. Taking the buckling strength of the unperforated plate into account for visualization purposes leads to more informative plots. Therefore, the critical loads obtained are divided by the buckling strength of the unperforated plate, which are obtained with ABAQUS and shown in Table 4.2. In the following sections, those ratios are shown in a 2-dimensional plot, again with the cutout origin coordinates on the x- and y-axes. They can now directly be compared against sensor-hole concept plots as presented in [7].

Table 4.2: Setup and results for the critical loads of the unperforated plate, for both loads cases. Obtained with ABAQUS.

Element		Critical Load Multiplier	
Type	Number	Load Case 1	Load Case 2
S8R5	40 000	759.20	1768.5

[5] https://www.mathworks.com/help/signal/ref/sgolay.html

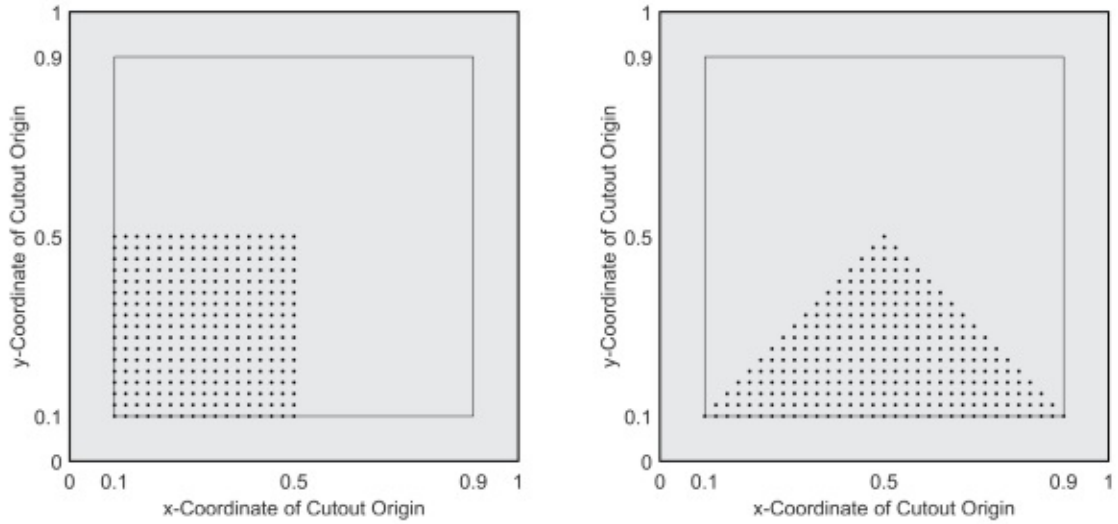


Fig. 4.1: Point grid discretization with $\delta = 0.025$ of the region $\mathcal{A}$ for LC1 (left) and LC2 (right). Only a quarter of the plate is discretized to utilize the respective symmetry of the results.

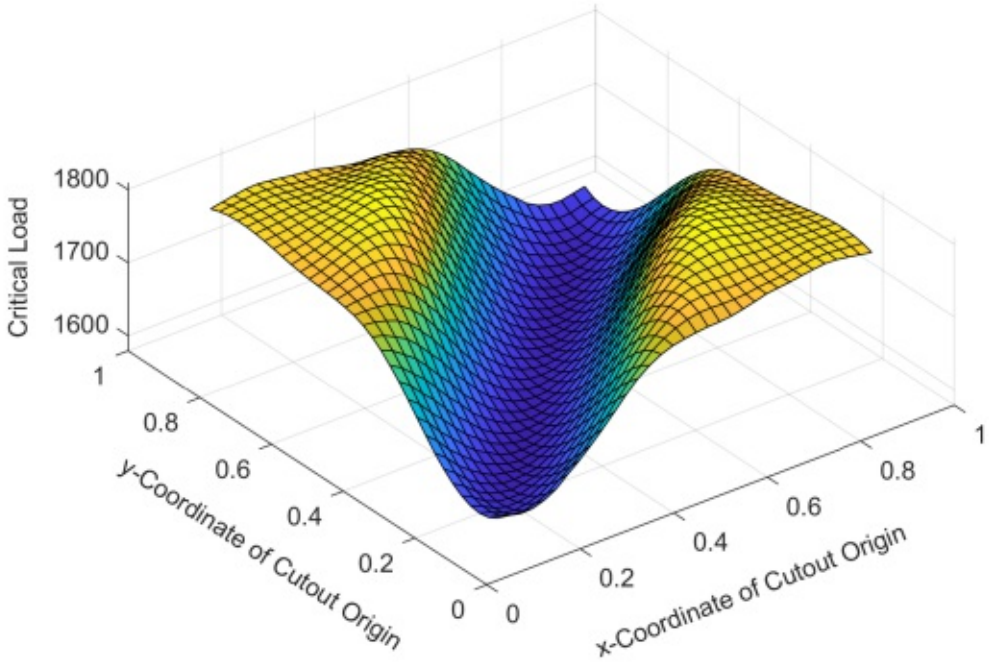


Fig. 4.2: Function of buckling loads in terms of the cutout origin coordinates for LC2: pure shear.

4.3 Results

The results from performed analysis and chosen visualization technique are shown and discussed in this section. Multiple conclusion can already be drawn, that hold for both load cases:

- As expected, and already elaborated above, the resulting critical loads are double-symmetric, see Figs. 4.3 and 4.5. This was proven with a full analysis first and then utilized by performing the analysis on only a quarter of the plate; reducing computational cost to one fourth.
- It is confirmed that there exist regions on this plate, where the chosen cutout leads to an increase in buckling strength. These regions, that are highlighted by a white isoline, again see Figs. 4.3 and 4.5, also qualitatively coincide with results from the sensor-hole concept, compare Figs. 4.4 and 4.6. Quantitatively equal results were not expected from the beginning, due to the finite cutout size used in this theses compared to the infinitesimally small hole in [7].
- When visually comparing results from this thesis to results from the sensor-hole concept it is important to note the difference in axes limits. A sensor-hole can theoretically be cut into the very edge of the structure, and the results can be shown for the complete plate area $\mathcal{A}' = [0.0, 1.0] \times [0.0, 1.0]$. Due to the arguments introduced above, this is not possible for the results obtained with the FCM-approach, resulting in results shown for the area $\mathcal{A} = [0.1, 0.9] \times [0.1, 0.9]$.

4.3.1 Load Case 1: Uniaxial Compression

For LC1, the FCM-results are shown in Fig. 4.3. A cutout location near the loaded edge, close to its midpoint, appears beneficial regarding buckling strength, increasing the buckling strength by up to 1%. At the same time, a cutout location at the midpoint of the plate is disadvantageous for the buckling strength, which is decreased by up to 5%. This results in a final spread of buckling loads of 6% for this load configuration.
In comparison with the sensor-hole concept results, depicted in Fig. 4.4, it must be noted that the area, where the ratio in buckling strength is greater than one, appears to be reduced. This however is considered a logical effect when increasing the cutout size, as done in this work.

4.3.2 Load Case 2: Pure Shear

For LC2, the FCM-results are shown in Fig. 4.5. Cutout locations in the upper-left and lower-right of the plate are beneficial regarding buckling strength, with an increase of up to 2%. Cutouts located along the main diagonal however significantly reduce the critical load by up to 10%, resulting in a total spread of 12% for this load configuration. It appears that a plate under pure shear loading is more sensitive in regards to the cutout position, compared to the uniaxially compressed plate.
Again in comparison with the sensor-hole concept, both beneficial and disadvantageous locations for cutouts match qualitatively. Increased hole size again results in reduced overall areas, where the ratio is greater than one. Interestingly however, the FCM-approach predicts higher critical loads in the corresponding corners of the plate than the sensor-holes would suggest.

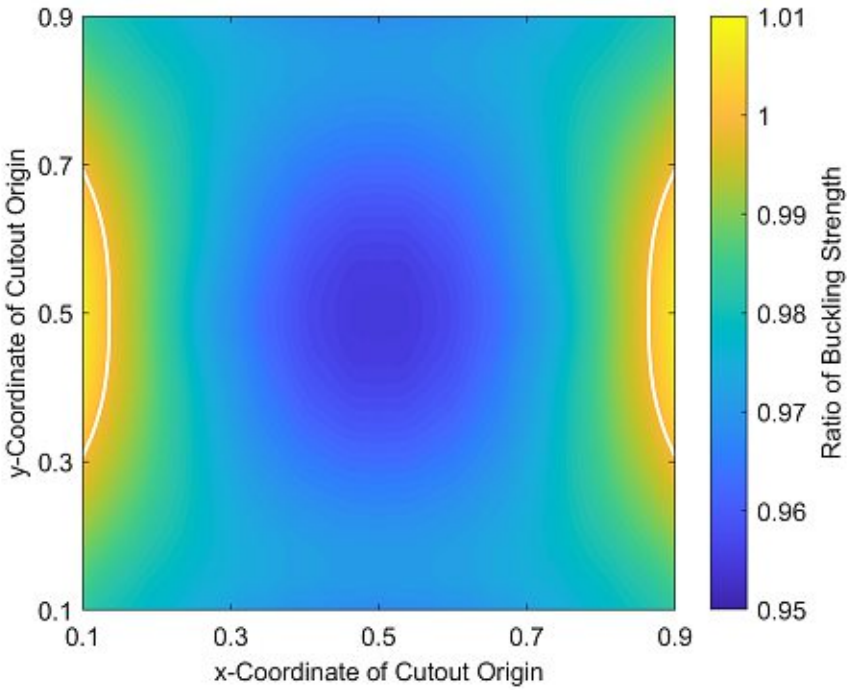


Fig. 4.3: Function of normalized buckling loads in terms of the cutout origin coordinates for LC1: uniaxial compression.

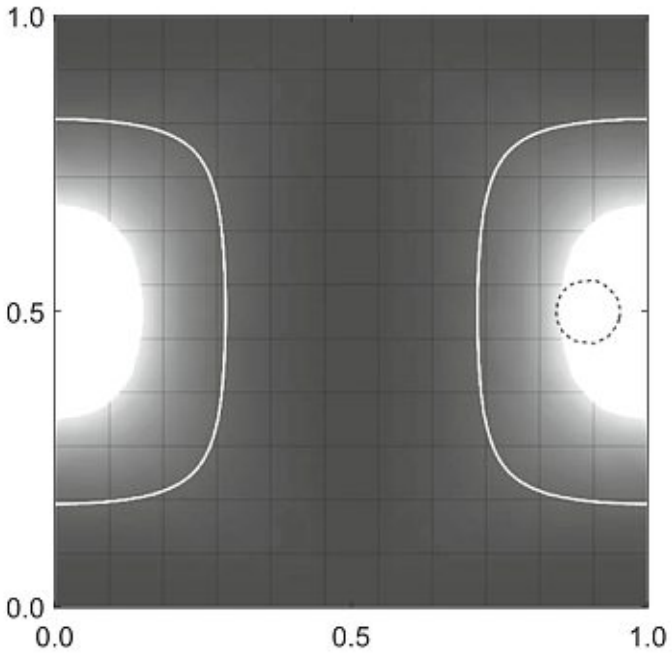


Fig. 4.4: Function of normalized buckling loads obtained with the sensor-hole concept for LC1: uniaxial compression [7]. The dashed circle denotes a cutout position that was additionally analyzed in the original paper.

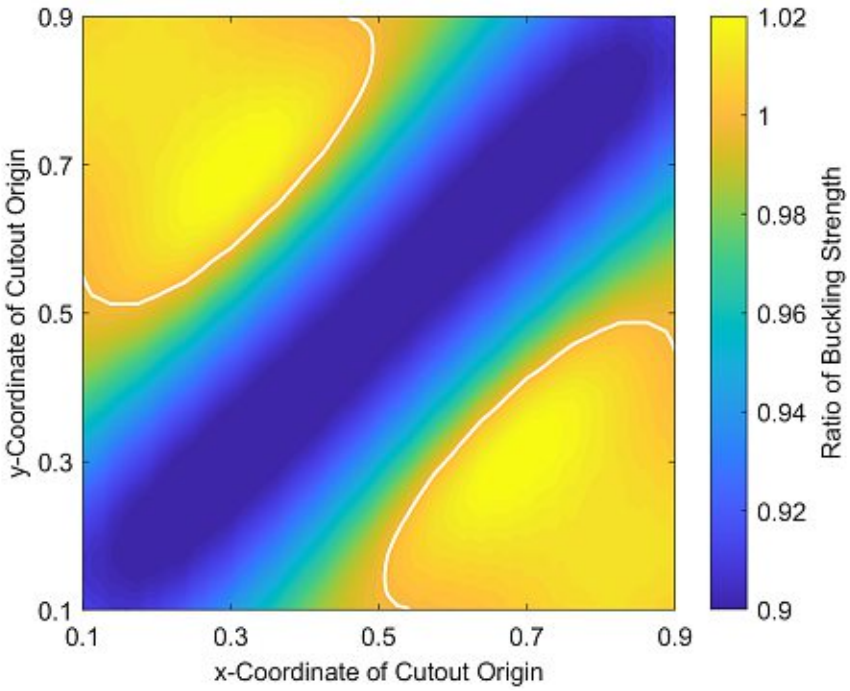


Fig. 4.5: Function of normalized buckling loads in terms of the cutout origin coordinates for LC2: pure shear.

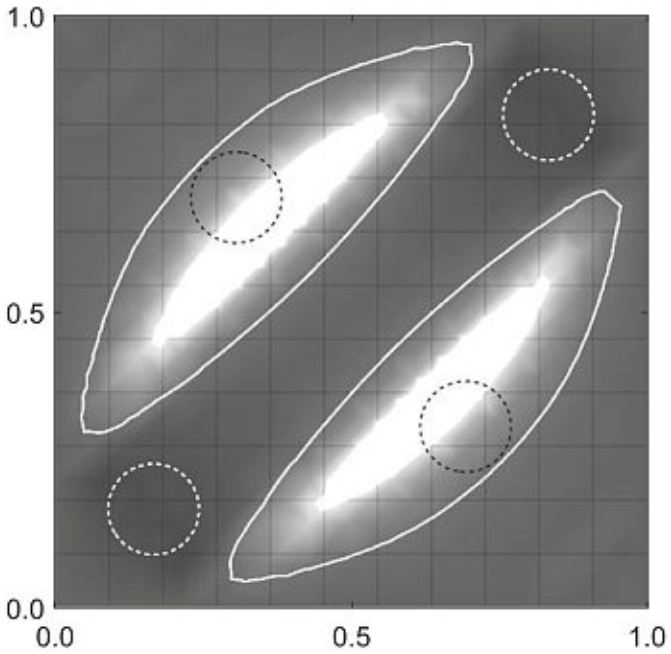


Fig. 4.6: Function of normalized buckling loads obtained with the sensor-hole concept for LC2: pure shear [7]. Dashed circles denote cutout positions that were additionally analyzed in the original paper.

Chapter 5

Conclusion & Outlook

5.1 Summary

The numerical analyses carried out in this thesis contribute to the broad topic of stability theory in structural mechanics in practice. The main focus was on implementing the rather new methodology of FCM in combination with plate problems. In this context, the key research question of finding the most advantageous position of a hole in a plate in regards to the plate's buckling strength could be answered with success.

The proposed approach proved its efficiency by overcoming the need for multiple discretizations, as usually needed in FEM for such optimization problems. In fact, only a single mesh is needed for the extended domain of the FCM. Due to its simple shape, BFS-shape functions could be applied, contributing all their beneficial properties. The main disadvantage restricting their application in traditional FEM is the requirement of continuous coordinate lines; it is of minor relevance in FCM. This makes BFS-elements very attractive in this scope of application.

Fundamental concepts of linear elasticity, plate theory, FEM and FCM were introduced in Chapter 2. Chapter 3 describes the implementation process, combining all of the above mentioned concepts in practical application. After proving the functionality of the code separately for a problem of linear elasticity and plate theory, they are coupled by the plane stress distribution for the buckling problem. A key aspect for efficiency and convergence is the correct computation of stresses in integration points, as the integration point location is not pre-determined in FCM. Ensuring the robustness of solving the buckling problem with an exemplary convergence analysis in Section 3.2, the optimization could be performed as planned in Chapter 4. It yields satisfactory results and answers the question of the best position for a hole; it is also in accordance with numerical results from commercial FEM-software as well as semi-analytical results from research papers.

This thesis also confirms the non-obvious statement, that well-placed cutouts can increase the buckling strength of plates. To emphasize present thesis' contribution to engineering practice, this effect can for example be used in the design of lightweight structures, where an increase in strength by reduction of structural mass is a highly desirable opportunity.

5.2 Future Considerations

Within the direct scope of this master's thesis, future work could further apply presented methodology on more plate problems. For example, different boundary conditions and load scenarios can be analyzed, increasing the plane stress distribution's complexity. Its influence on the stability behavior may be exciting to analyze. Additionally, the cutout geometry can be changed freely, both in shape and size. An interesting research question could be the identification of the optimum shape of a cutout with prescribed surface area, or the stability behavior of plates with multiple holes.

Showing the performance of FCM in combination with BFS shape functions, there are several opportunities to widen the applicability of this approach. Quite naturally an extension to nonlinear buckling analysis comes to mind, where also pre-buckling deformation and post-buckling behavior can be studied. Additionally, this approach is by no means restricted to plate problems. The incorporation of FCM with shell elements would greatly broaden the possible applications, as already shown in a recent research paper [5]. BFS-shape functions are also compatible with shells elements, which would make such an implementation a direct successor of this thesis. Finally, the usage of BFS-shape functions in general is greatly simplified as its restriction can be elegantly avoided in combination with FCM. This may lead to further possible applications for them, as well as valuable new insights.

Bibliography

[1] L. Euler. “Additamentum I. De Curvis Elasticis”. In: *Methodus inveniendi lineas curvas maximi minimive proprietate gaudentes*. Marc-Michel Bousquet, 1744.

[2] S. P. Timoshenko and J. M. Gere. *Theory of Elastic Stability*. 2nd ed. McGraw-Hill, 1961.

[3] M. Eslami. *Buckling and Postbuckling of Beams, Plates, and Shells*. 2nd ed. Springer, 2018.

[4] F. G. Rammerstorfer. “Buckling of elastic structures under tensile loads”. In: *Acta Mech* 229 (2018), pp. 881–900.

[5] Y. Guo, H. Do, and M. Ruess. “Isogeometric stability analysis of thin shells: From simple geometries to engineering models”. In: *International Journal for Numerical Methods in Engineering* 118:8 (2019), pp. 433–458.

[6] A. Bagchi. “Linear and Nonlinear Buckling of Thin Shells of Revolution”. In: *Trends in Applied Sciences Research* 7 (2012), pp. 196–209.

[7] J. B. Gracia and F. G. Rammerstorfer. “Increase in buckling loads of plates by introduction of cutouts”. In: *Acta Mechanica* 230 (2014), pp. 2873–2889.

[8] A. I. Lurie and A. Belyaev. “Deformation of a continuum”. In: *Theory of Elasticity*. Springer, 2005, pp. 77–123.

[9] Y. Vetyukov and M. Krommer. “Höhere Festigkeitslehre”. Lecture notes. Vienna University of Technology, WS 2020.

[10] J. Parvizian, A. Düster, and E. Rank. “Finite cell method: h- and p-extension for embedded domain problems in solid mechanics”. In: *Computational Mechanics* 41 (2007), pp. 121–133.

[11] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. “11 - Plate bending approximation: Thin (Kirchhoff) plates and C_1 continuity requirements”. In: *The Finite Element Method Set, Volume 2*. 6th ed. Butterworth-Heinemann, 2005, pp. 323–381.

[12] B. Szabó and I. Babuska. “An Outline of the Finite Element Method”. In: *Introduction to Finite Element Analysis*. Wiley and Sons, 2011. Chap. 2, pp. 17–78.

[13] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. “2 - A direct physical approach to problems in elasticity: Plane stress”. In: *The Finite Element Method Set, Volume 1*. 6th ed. Butterworth-Heinemann, 2005, pp. 19–53.

[14] R. de Borst, C. M., J. Remmers, and C. Verhoosel. *Nonlinear Finite Element Analysis of Solids and Structures*. 2nd ed. Wiley and Sons, 2012.

[15] Dassault Systèmes Simulia Corp. *ABAQUS 2016 Theory Guide*. Providence, RI, USA, 2016.

[16] F. Bogner, R. Fox, and L. Schmit. “The generation of interelement compatible stiffness and mass matrices by the use of interpolation formulas”. In: *1st Conf. Matrix Methods in Structural Mechanics (Wright Petterson Air Force Base, Ohio)* (1966), pp. 397–443.

[17] Y. Vetyukov. “Finite element modeling of Kirchhoff-Love shells as smooth material surfaces”. In: *Journal of Applied Mathematics and Mechanics* 1–2 (2014), pp. 150–163.

[18] J. Valdman. “MATLAB Implementation of C1 Finite Elements: Bogner-Fox-Schmit Rectangle”. In: *Parallel Processing and Applied Mathematics*. Springer International Publishing, 2020, pp. 256–266.

[19] Y. Vetyukov. *Nonlinear Mechanics of Thin-Walled Structures. Asymptotics, Direct Approach and Numerical Analysis*. Springer, 2014.

[20] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. “14 - Adaptive finite element refinement”. In: *The Finite Element Method Set (Sixth Edition)*. 6th ed. Butterworth-Heinemann, 2005, pp. 500–524.

[21] M. Petö, F. Duvigneau, and S. Eisenträger. “Enhanced Numerical Integration Scheme based on Image-Compression Techniques: Application to Fictious Domain Methods”. In: *Advanced Modeling and Simulation in Engineering Sciences* 7:21 (2020).

[22] S. Duczek and U. Gabbert. “Efficient integration method for fictitious domain approaches”. In: *Computational Mechanics* 56 (2015), pp. 725–738.

[23] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. *The Finite Element Method Set*. 6th ed. Butterworth-Heinemann, 2005.

[24] B. Szabó, A. Düster, and E. Rank. “The p-Version of the Finite Element Method”. In: *Encyclopedia of Computational Mechanics*. Wiley and Sons, 2004.

[25] K. Koketsu, H. Fujiwara, and Y. Ikegami. “Finite-element Simulation of Seismic Ground Motion with a Voxel Mesh”. In: *Pure and Applied Geophysics* 161 (2004), pp. 2183–2198.

[26] Z. Yang, M. Ruess, S. Kollmannsberger, A. Düster, and E. Rank. “An efficient integration technique for the voxel-based finite cell method”. In: *International Journal for Numerical Methods in Engineering* 91.5 (2012), pp. 457–471.

[27] R. Fletcher. *Practical Methods of Optimization*. 3rd ed. Wiley and Sons, 2007.

[28] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes: The Art of Scientific Computing*. 3rd ed. Cambridge University Press, 2007.

[29] J. Lagarias, J. Reeds, M. Wright, and P. Wright. “Convergence Properties of the Nelder–Mead Simplex Method in Low Dimensions”. In: *SIAM Journal on Optimization* 9 (1998), pp. 112–147.