

Masterarbeit im Studiengang Bauingenieurwissenschaften
am Institut für Baumechanik

Technische Universität Graz

Dynamic Analysis of Two-Dimensional Frames Using the Numerical Assembly Technique

Thomas Kramer

Graz, 17. März 2021

Betreuer: Ass.Prof. Dipl.-Ing. Dr.techn. BSc Gfrerer Michael Helmut

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Graz, 17.03.2021

Date



Signature

Eidesstattliche Erklärung¹

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am 17.03.2021

Datum



Unterschrift

¹Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008; Genehmigung des Senates am 1.12.2008

Acknowledgements

First and foremost, I want to thank my supervisor, Ass.Prof. Dipl.-Ing. Dr.techn. BSc, Gfrerer Michael Helmut for his enormous support and expertise throughout the execution of my thesis. Even though he had a limited window of time while raising two wonderful kids, he still had enough energy to give me advice whenever I needed it.

I would also like to express my gratitude to my beloved fiancé. I want to thank her for her patience and understanding. She helped me not only academically, but also supported me as best she could outside of university matters.

My appreciation also extends to our families and friends, who motivated me during trying times. Thanks to their encouragement, I managed to carry on and stay focused. Without their help and support, I would not be where I am.

Abstract

Proposed is a MATLAB-based code to compute a wanted number of natural frequencies and mode shapes of any arbitrary two-dimensional frame structure. The considered frame can consist of several continuous beam elements. Common supports, hinges, and springs can be applied at the nodes. The computations are executed analytically and refer to the exact solutions of the homogenous differential equations specified on each elastic beam of the structure before the application of boundary and interface conditions. Hence, the discretization of a frame-system at the nodes is sufficient. The computation is restricted to the two-dimensional space considering longitudinal and transversal vibrations. The natural frequencies of structures can be obtained based on either the Euler-Bernoulli or the Timoshenko beam theory, and the corresponding mode shapes are plotted.

Especially, when the natural frequencies need to be determined exactly, as e.g. for sensitive technical equipment, robotic arms etc. this thesis can be of high relevance. The "Numerical Assembly Technique" (NAT) was used to determine the natural frequencies which relies on solving a system matrix that is a result of the general differential equation with enforced boundary conditions. The created code was tested on several structures and the implementation in MATLAB is described.

Keywords: Dynamic Analysis of Frames, Numerical Assembly Technique, NAT, Euler-Bernoulli Beam Theory, Timoshenko Beam Theory, Natural Frequencies, Mode Shapes

Zusammenfassung

Vorgestellt wird ein MATLAB-basierter Code zur Berechnung einer gewünschten Anzahl von Eigenfrequenzen und Eigenformen beliebiger zweidimensionaler Rahmenstrukturen. Der betrachtete Rahmen kann aus mehreren zusammenhängenden Balkenelementen bestehen. An den Knotenpunkten können gängige Lager, Drehgelenke und Federn angebracht werden. Die Berechnungen werden analytisch durchgeführt und beziehen sich auf die exakten Lösungen der homogenen Differentialgleichungen, die auf jedem elastischen Balken der Struktur vor der Aufbringung von Rand- und Übergangsbedingungen angegeben sind. Das Diskretisieren eines Rahmensystems an den Knotenpunkten ist daher ausreichend. Die Berechnung beschränkt sich auf die zweidimensionale Ebene und berücksichtigt longitudinale und transversale Schwingungen. Die Eigenfrequenzen der Strukturen können entweder nach der Euler-Bernoulli- oder der Timoshenko-Balkentheorie ermittelt werden und die entsprechenden Eigenformen werden grafisch dargestellt.

Insbesondere, wenn die Eigenfrequenzen exakt bestimmt werden müssen, wie beispielsweise bei empfindlichen technischen Geräten, Roboterarmen etc. kann diese Arbeit von hoher Relevanz sein. Zur Bestimmung der Eigenfrequenzen wurde die "Numerical Assembly Technique" (NAT) verwendet, die auf der Lösung einer Systemmatrix beruht, welche ein Resultat aus der allgemeinen Differentialgleichung mit erzwungenen Randbedingungen ist. Der erstellte Code wurde an mehreren Strukturen getestet und die Implementierung in MATLAB wird in dieser Thesis ausführlich beschrieben.

Schlagwörter: Dynamische Analyse von Rahmensystemen, Numerical Assembly Technique, NAT, Euler-Bernoulli Balkentheorie, Timoshenko Balkentheorie, Eigenfrequenzen, Eigenformen

CONTENTS

Index	i
List of Figures	iii
List of Tables	v
Nomenclature	vi
1. Introduction	1
2. Mechanical Models	5
2.1. Transversal Vibrations	5
2.2. Longitudinal Vibrations	13
3. Application	15
3.1. Structural Definitions	15
3.2. Differential Equations	15
3.3. Boundary Conditions	17
3.4. System of Equations	21
3.5. Natural Frequencies	23
3.6. Mode Shapes	27
4. Test Examples	31
4.1. Single-Span Beam	31
4.2. Clamped-Pinned Beam	37
4.3. Cantilever Beam	41
4.4. Two-Beam Frame	46
5. Practical Examples	49
5.1. Multi-Storey Frame	49
5.2. Railway Bridge	51
6. Implementation	53
6.1. computeNaturalFrequencies.m	55
6.2. inputFile.m	64
6.3. BCs.m	66
6.4. subSoE.m	71
6.5. plotFile.m	71
7. Conclusion	79
A. Different Approach for the General Solution	81
B. The Behaviour of the Eigenvalues for a Timoshenko Beam	83
References	87

LIST OF FIGURES

1.1.	Arbitrary frame vibration problem.	3
2.1.	The equilibrium condition between external and internal forces (left), and the vertical deflection w and rotation ψ of a small element (right).	5
2.2.	The composition of a 'sandwich beam'.	10
2.3.	The bending and shear deformation of a Timoshenko beam.	10
2.4.	The equilibrium condition for longitudinal forces on a small element.	13
3.1.	The resulting kinetic force S (left) and the total deformation \bar{U} (right) at the frame node $k_{1,0}$	16
3.2.	The decomposed kinetic force (left) and the total deformation (right) at the frame node $k_{1,0}$	16
3.3.	The equilibrium conditions at each frame node $k \in N_\tau$ (bottom), applied at an example frame (top).	17
3.4.	Attachment of a roller in a slanted position.	18
3.5.	Attachment of a longitudinal spring.	19
3.6.	Attachment of a rotational spring.	19
3.7.	Plot of the characteristic function for a Euler-Bernoulli cantilever beam in the range of $\omega = 0 - 10 \text{ rad/s}$	23
3.8.	One iteration-step of the regula-falsi method.	25
3.9.	The matrix of eigenvectors $[\mathbf{v}_i]$ and eigenvalues $[\lambda_i]$ for one specific natural frequency ω_i	27
3.10.	The matrix of eigenvectors for a cantilever beam evaluated at the first eigenfrequency ω_1 , highlighting the j^{th} column containing the lowest eigenvalue on the main diagonal.	27
3.11.	The matrix of eigenvectors for a cantilever beam evaluated at the first eigenfrequency ω_1 , highlighting the j^{th} column responsible for the first transversal eigenmode.	28
3.12.	The contribution of the integration constants and bearing variables on the first transversal mode shape of a cantilever beam.	28
3.13.	The first transversal mode shape of a cantilever beam.	29
4.1.	Characteristic function for a single-span beam based on the Euler-Bernoulli beam theory.	34
4.2.	The first four eigenmodes of a single-span beam.	35
4.3.	The eigenmodes ϕ_i with $i = \{1, ..4\}$ of a single-span beam in one plot.	35
4.4.	The quantitative internal forces occurring at the natural frequencies 1-4.	36
4.5.	Characteristic function for a single clamped-pinned beam.	38

4.6.	The first four eigenmodes of a clamped-pinned beam.	39
4.7.	The eigenmodes $\phi_i, i = \{1, 2, ..4\}$ of a clamped-pinned beam in one plot. .	39
4.8.	The quantitative internal forces of a clamped-pinned beam occurring at the natural frequencies 1-4.	40
4.9.	Characteristic function for a cantilever beam.	43
4.10.	The eigenmodes of a cantilever beam.	44
4.11.	The eigenmodes $\phi_i, i = \{1, 2, ..5\}$ of a cantilever beam in one plot.	44
4.12.	The quantitative internal forces of a cantilever beam occurring at the natural frequencies 1-4.	45
4.13.	Characteristic function $f(\omega)$ for the two-beam frame.	47
4.14.	Characteristic function $f(\omega)$ on a logarithmic scale.	47
4.15.	The mode shapes $\phi_i, i = \{1, 2, 3\}$ of the two-beam frame.	48
5.1.	The sketch of the multi storey frame.	49
5.2.	The first lateral eigenmode of the multi storey frame.	50
5.3.	The sketch (top) and the first mode shape (bottom) of the railway bridge. .	51
6.1.	The hierarchy of code files.	53
6.2.	The computation process to calculate the natural frequencies.	54
6.3.	The system plot of the multiple beam frame.	56
6.4.	The first two eigenmodes of the multiple beam frame.	78
A.1.	The difference of Ansatz functions for the general solution. The hyperbolic- (left) and exponential approach (right).	81
A.2.	The behaviour of the conditioning number for the system matrix.	82
B.1.	The eigenvalues for the frequency range I.	84
B.2.	The eigenvalues for the frequency range II.	85
B.3.	The eigenvalues for the frequency range III.	85

LIST OF TABLES

1.1. References using the NAT.	2
3.1. A list of the different support types and its support conditions.	20
3.2. A list of the different springs and hinges and its support conditions.	21
3.3. Frequency functions for transversal vibrations.	26
3.4. Frequency functions for longitudinal vibrations.	26
4.1. The natural frequencies of the single-span beam in [rad/s].	34
4.2. The natural frequencies of the clamped-pinned beam in [rad/s].	39
4.3. λ_i and ω_i for a cantilever beam.	41
4.4. The natural frequencies of the cantilever beam in [rad/s].	43
4.5. The natural frequencies of the two-beam frame in [rad/s].	48
5.1. The first natural frequency of the multi-storey frame in [rad/s].	50
5.2. The first natural frequency of the railway bridge in [rad/s].	51
6.1. The initial configuration of bearings and springs.	65
6.2. The first two natural frequencies of the multi-beam frame in [rad/s].	78

Nomenclature

m	frame member	t	time
λ_i	polynomial roots	ϕ_i	i^{th} mode shape
$C_{j,i}$	integration constants $j=1,2,\dots,6$	x_i, y_i	local system coordinates
E_m	Young's modulus of elasticity	ω	circular frequency
G_m	Shear modulus of rigidity	ψ	slope due to bending
I_m	Moment of Inertia	W	bending deflection, frequency domain
A_m	cross-section area of the beam	w	bending deflection, time domain
κ	shear deformation shape coefficient	U	long. deformation, frequency domain
c	spring stiffness	u	long. deformation, time domain
L_m	length of the beam	ρ_m	mass density of the beam material
m_m	mass of the beam	μ	mass distribution
n_m	number of beams	ξ	non-dimensional beam length $(x/L)_m$
V	shear force	M	rotational moment
N	axial force	Θ	rotational mass
A_s	effective cross-section shear area	q	loading force
∂	partial differentiation	c	wave propagation velocity
τ	set of frame members	N_τ	frame nodes
\mathbf{v}	matrix of eigenvectors	λ	matrix of eigenvalues
$\{\}$	vector notation	$[\]$	matrix notation
x	subscript for x-component	z	subscript for z-component
y	subscript around y-axis	n	number of eigenfrequencies
B	bending stiffness	S	shear stiffness
$f(\omega)$	frequency function	X, Y	global coordinate system
EBB	Euler-Bernoulli beam	TIB	Timoshenko beam
n_f	number of natural frequencies	i	subscript for i^{th} eigenvalue

1. INTRODUCTION

The deformation of an elastic body can only be described approximately when using either the method of Multi-Body Systems (MBS) or the Finite Element Method (FEM). When modelling the elastic continuum in a refined and detailed way through infinitesimally small elements, the body shows an infinite number of degrees of freedom. Its deformation can be described locally by partial differential equations, which can only be solved by applying intermediate and boundary conditions. This leads to an infinite-dimensional eigenvalue problem that holds the exact description of the body's deformation. Narrowing down the amount of elements to a finite number defining the system, where constant parameters hold within the elements, allows one to determine the exact solutions for a system of elements [1,2].

A significantly high number of mechanical and structural engineering problems can be modelled by an assembly of beams with attached supports, hinges, and springs such as building structures but also robotic arms or any mechanical mechanism. Especially structural features where the dynamic behaviour is of major interest, the natural frequencies play an important role. Thus, having an efficient tool to determine those natural frequencies, combined with the corresponding mode shapes, helps engineers perform high-quality work. In many cases, the structure is modelled in finite element software packages that are based on an approximation of the exact solution and requires a laborious discretization process. Working with methods based on the analytical solution is, therefore, less time-consuming and the results are more accurate. The Numerical Assembly Technique (NAT) [3–9] counts as such a method.

In several publications, natural frequencies and mode shapes were calculated analytically. Gasch and Knothe [4] described the analytical determination of eigenfrequencies on simple structures such as single-span beams and cantilever beams in their work. The same applies to Dinkler [2], where he explained the fundamentals of structural dynamics. Klanner and Ellermann [3] compiled a comprehensible overview of analytic methods for dynamic analysis, which comprises several techniques. These include the Numerical Assembly Technique (NAT), the Transfer Matrix Method (TMM) [10–14], the Dynamic Stiffness Method (DSM) [15, 16], the Green Function Method (GFM) [17–21], the Generalized Function Approach (GFA) [22–24], and the Modal Superposition Method (MSM) [25]. The latter describes a method to analyse forced vibrations and can be applied to the other mentioned methods. Nonetheless, the main focus in Klanner's work is the utilization of the Numerical Assembly Technique on a multiple-stepped Euler-Bernoulli beam with arbitrarily distributed force or moment loading.

Similar publications exist, such as the dynamic analysis on multiple-stepped Timoshenko beams with several applied concentrated masses or spring-mass systems, using the Numerical Assembly Technique, for instance, in Faraghaly's and El-Sayed's work [5]. Several

other researchers, such as Hsien-Yuan Lin [7], Jee-Ray Wang, et al. [8], and Der-Wei Chen [9], have used their expertise to find the exact solution of natural frequencies for Timoshenko multi-stepped beams with attached concentrated masses and spring-mass systems. Yusuf Yesilce and Oktay Demirdag extended the research by observing the influence of axial forces on such beams [6]. A summarized list of papers referring to the determination of natural frequencies using the Numerical Assembly Technique can be found in Table 1.1.

Table 1.1.: References using the NAT.

No.	Title	Author
[9]	The exact solutions for the natural frequencies and mode shapes of non-uniform beams carrying multiple various concentrated elements	Chen, Der-Wei
[5]	Exact free vibration of multi-step Timoshenko beam system with several attachments	Farghaly, SH and El-Sayed, TA
[4]	Strukturdynamik-Band 2: Kontinua und ihre Diskretisierung	Gasch, R. and Knothe, K
[3]	Steady-state harmonic vibrations of multiple-stepped Euler-Bernoulli beams under arbitrarily distributed loads carrying any number of concentrated elements	Klanner, M. and Ellermann, K
[7]	On the natural frequencies and mode shapes of a multi-span Timoshenko beam carrying a number of various concentrated elements	Lin, Hsien-Yuan
[8]	Free vibration analysis of a Timoshenko beam carrying multiple spring-mass systems with the effects of shear deformation and rotary inertia	Wang, Jee-Ray and Liu, et al
[6]	Effect of axial force on free vibration of Timoshenko multi-span beam carrying multiple spring-mass systems	Yesilce, Yusuf and Demirdag, Oktay

In the mentioned publications, the beam theories were mostly applied to basic structural models such as single-span or cantilever beams. These observations led to the formulation of a MATLAB-based code that calculates the natural frequencies in a wished frequency range and its corresponding mode shapes of any arbitrary, two-dimensional frame structure Figure 1.1. It should be possible to choose between the Euler-Bernoulli and Timoshenko beam theory to get an idea about the influence of shear deformation and rotational inertia. Also, a large selection of different supports and hinges, in addition to rotational and translational springs, are provided. To easily interpret computational results, the output is transformed into standalone L^AT_EX-files. This is practical when visualizing the mode shapes and incorporating them in reports.

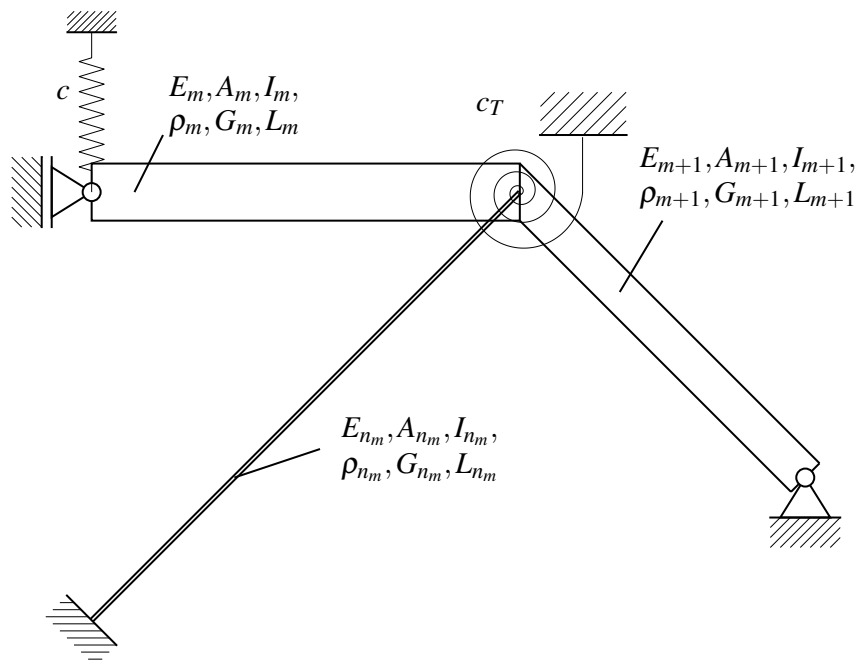


Figure 1.1.: Arbitrary frame vibration problem.

This thesis is divided into seven chapters. After the introduction as the first chapter, the mechanical models, such as the derivation for the longitudinal, as well as the transversal vibrations are presented in Chapter 2. Moreover, the terms are also simplified in order to differentiate the Euler-Bernoulli beam theory from the Timoshenko beam theory. In Chapter 3, the application of the derived differential equations to structures is discussed. To do so the main calculation processes is explained after a clear definition of the structure. Knowing the way of how the calculation process works, the procedure was applied on simple test-examples in Chapter 4. The computational results were always compared to handwritten calculations to assure the liability of the code. Throughout the examples, the different results of both beam theories were studied. In Chapter 5, the code is used for the dynamic analysis of more practical examples. Furthermore, a closure insight into the implementation of the calculation process in the MATLAB code is given in Chapter 6. The computational steps are denoted to comprehend and retrace the code easier from the reader's point of view. Finally, the work is concluded in Chapter 7 with the hope of new research findings that could help to progress the computational way of finding natural frequencies abroad from using the finite element approach.

2. MECHANICAL MODELS

The mechanical models are split up into transversal and longitudinal vibrations, as both are considered in the two-dimensional space. Thus, the derivation of both is given on straight and uniform beams. Within an element the governing equation of motion holds. Meaning the material, and the geometrical parameters of this beam element, are constant over the length and supports can only be applied on either the starting or end node of a beam element.

2.1. Transversal Vibrations

The focus is on deriving the general differential equation of motion, meaning that this equation will be served to obtain the eigenfrequencies of beams with various bearings. To stay as general as possible, first, we assume that the beam is shear elastic and that the cross-section could vary along the coordinate x . Whereas, m and Θ_y define the translational and rotational inertial masses, V and M the internal shear force and bending moment, A and I the cross-sectional area and moment of inertia, and q and ρ the externally applied force and the material density, respectively.

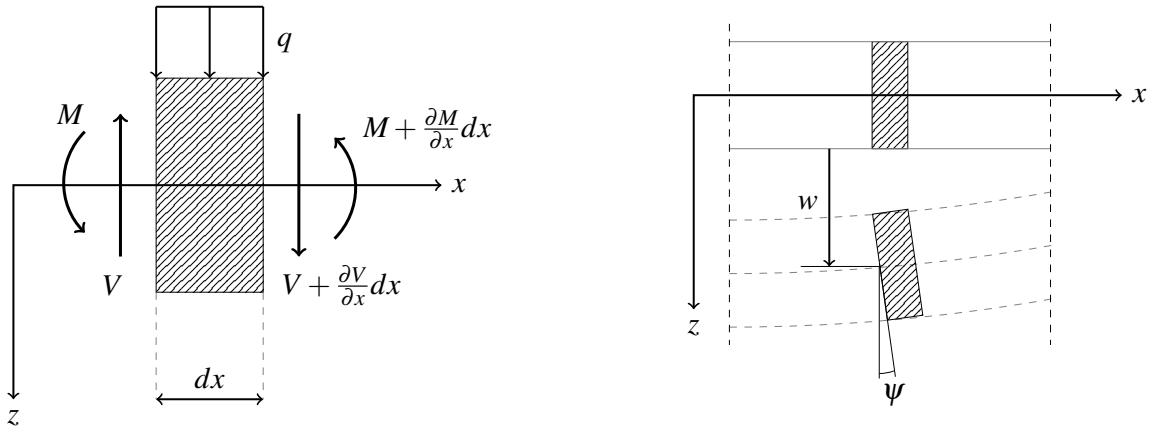


Figure 2.1.: The equilibrium condition between external and internal forces (left), and the vertical deflection w and rotation ψ of a small element (right).

Throughout the deformation, a very small element of the beam can move in z -direction noted as $w_{(x,t)}$ (2.1), as well as rotate around the y -axis, noted as $\psi_{(x,t)}$ (2.2). Therefore the equations can be set up considering the equilibrium condition, see Figure 2.1, as

$$dm \ddot{w} = -V + \left(V + \frac{\partial V}{\partial x} dx \right) + q dx \quad \rightarrow \quad \rho A \ddot{w} = V' + q \quad (2.1)$$

and

$$d\Theta_y \ddot{\psi} = -M + \left(M + \frac{\partial M}{\partial x} dx \right) - V dx \quad \rightarrow \quad \rho I \ddot{\psi} = M' - V, \quad (2.2)$$

referring to the equilibrium condition on a small element of the beam in vertical direction and rotation respectively, see Figure 2.1. Therefore, $()'$ denotes the derivative with respect to the axial coordinate x and $()$ the time derivative. In addition we assume the linear elastic constitutive relation for the bending moment,

$$M = EI\psi', \quad (2.3)$$

as well as for the shear force

$$V = GA_s(w' + \psi). \quad (2.4)$$

Considering shear deformations, the actual cross-sectional area A is reduced by the shear coefficient $\bar{\kappa}$ to A_s , bearing the shear force. Thus, it holds $A_s = \bar{\kappa}A$. The shear coefficient $\bar{\kappa}$ indicates the effective shear area due to a non-constant distribution of the shear stresses over the cross section and depends therefore on the form and dimensions of the cross-section. For beams with a constant rectangular cross section it holds that $\bar{\kappa} = 5/6$.

However, four equations exist to solve for the unknowns M, V, w and ψ . The theory based on those equations is called the Timoshenko beam theory, meaning that shear deformation and the rotational inertia are respected and will be further analysed in Section 2.1.2. M and V can be eliminated by applying equation (2.3) and (2.4) into (2.1) and (2.2). Therefore, we can obtain the two linked differential equations of second order

$$\rho A \ddot{w} - [GA_s(w' + \psi)]' = q, \quad (2.5)$$

and

$$\rho I \ddot{\psi} - [EI\psi']' + GA_s(w' + \psi) = 0. \quad (2.6)$$

The differential equations (2.5) and (2.6) can be interpreted as the general base equations for further simplifications discussed in the Section 2.1.1 and Section 2.1.2. To solve the

differential equations, boundary conditions need to be determined from bearing situations as well as other properties as they will be mandatory to form the integration constants.

Before specifying the two different beam theories, the differential equations should be simplified for a constant cross-section along the x - axis. Therefore, the equation (2.3) can be written as

$$M = EI(w' + \psi)' - EIw'' \quad (2.7)$$

and the equation (2.2) as

$$\rho I(w' + \psi)'' - \rho I\dot{w}' - M' + V = 0. \quad (2.8)$$

After forming the derivatives with respect to x , the combined differential equation of motion for a beam with a constant cross section can be obtained as

$$EIw^{IV} + \rho A\ddot{w} - \rho I\left(1 + \frac{EA}{GA_s}\right)\ddot{w}'' + \rho A\frac{\rho I}{GA_s}\ddot{w} = q + \frac{\rho I}{GA_s}\ddot{q} - \frac{EI}{GA_s}q''. \quad (2.9)$$

2.1.1. Euler-Bernoulli beam theory

The main idea of the Euler-Bernoulli beam theory is based on the simplifications of an infinite high shear stiffness ($GA_s \rightarrow \infty$) and zero rotational inertia ($\rho I \rightarrow 0$).

Those assumptions simplify the differential equations derived in Section 2.1. The equation (2.2) changes to the equilibrium condition $M' - V = 0$ and equation (2.4) is simplified to $w' + \psi = 0$. The four obtained equations

$$\rho A\dot{w} = V' + q, \quad M' = V, \quad EI\psi' = M, \quad w' = -\psi \quad (2.10)$$

can then be implemented to obtain the resulting Euler-Bernoulli equation of motion

$$(EIw'')'' + \rho A\dot{w} = q, \quad (2.11)$$

which simplifies in case of $EI = const.$ to

$$EIw^{IV} + \rho A\dot{w} = q. \quad (2.12)$$

To obtain results for the eigenfrequencies and the eigenmodes of the structure the non-trivial solutions for the eigenvalues of the differential equation need to be found. This can be either done by solving the differential equation remaining in the time domain or by transferring the differential equation to the frequency domain. In this thesis the calculations are done in the frequency domain. Nevertheless, both possibilities will be derived.

Free vibrations define the dynamic displacements of a structure without an excitation force affecting the vibrations, meaning that the right-hand side of the equation is zero as the outer load $q = 0$. Still, the system needs to vibrate. Therefore, a starting deflection needs to be applied.

It is already known that the equation of motion is

$$\frac{\partial^4 w}{\partial x^4} + \frac{\rho A}{EI} \frac{\partial^2 w}{\partial t^2} = 0. \quad (2.13)$$

To solve the equation an Ansatz for $w_{(x,t)}$ needs to be found. Thus, the Ansatz for harmonic oscillations ωt with a phase angle φ was chosen for the displacement in the time domain, so that

$$w_{(x,t)} = W_{(x)} \cos(\omega t - \varphi) \quad (2.14)$$

Cancelling out the $\cos()$ -terms leads to the differential equation in the frequency domain

$$\frac{d^4 W}{dx^4} - \kappa^4 W = 0. \quad \text{with} \quad \kappa^4 = \omega^2 \frac{\rho A}{EI} \quad (2.15)$$

For the complex displacement $W_{(x)}$ in the frequency domain, the Ansatz is

$$W_{(x)} = e^{\lambda x}. \quad (2.16)$$

After dropping the exponential terms, the fourth-degree polynomial is

$$\lambda^4 - \kappa^4 = 0. \quad (2.17)$$

When solving the characteristic polynomial fourth order (2.17), the four roots of the eigenvalue problem are $\lambda_1 = +\kappa$, $\lambda_2 = +i\kappa$, $\lambda_3 = -\kappa$, and $\lambda_4 = -i\kappa$. Where

$$\kappa = \sqrt{\omega} \sqrt[4]{\frac{\rho A}{EI}}. \quad (2.18)$$

A common general solution respecting those eigenvalues is the straight trigonometric approach

$$W_{(x)} = C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 \cosh(\kappa x) + C_4 \sinh(\kappa x). \quad (2.19)$$

In this thesis, a slightly different approach was chosen so that the general solution has the form

$$W_{(x)} = C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 e^{\kappa(x-L)} + C_4 e^{-\kappa x}. \quad (2.20)$$

Using the solution (2.20) guaranties that the amplitude of each function term remains smaller or equal to 1 within the segment span ($0 < x < L$), whereas the amplitude of the common approach Equation (2.19) rises as the beam length and/or the observed frequency range increases. The proposed solution Equation (2.20) appears to be more stable especially for higher frequencies than the commonly used one in (2.19). More details can be obtained in Appendix A.

However, Equation (2.20) implemented in the Ansatz (2.14) forms the general solution in the time domain

$$w_{(x,t)} = [C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 e^{\kappa(x-L)} + C_4 e^{-\kappa x}] \cos(\omega t - \varphi). \quad (2.21)$$

The internal forces for the Euler-Bernoulli beam are defined as

$$M = -EI \frac{\partial^2 W}{\partial x^2} \quad \text{and} \quad V = \frac{\partial M}{\partial x}. \quad (2.22)$$

2.1.2. Timoshenko beam theory

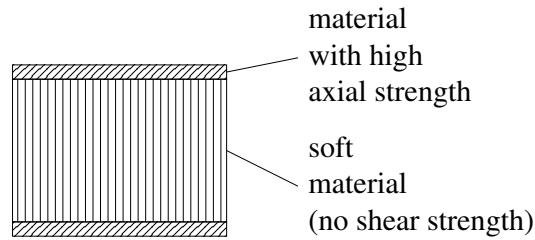
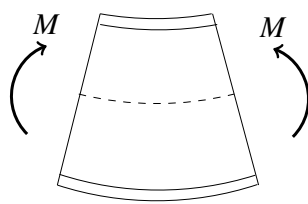
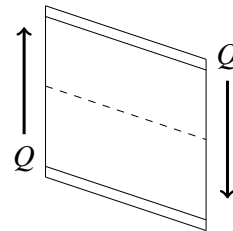


Figure 2.2.: The composition of a 'sandwich beam'.

The theory of a Timoshenko beam is based on a model that considers shear deformations and rotational inertia. Meaning that the beam includes finite shear strength $0 < G < \infty$ and rotational inertia $\rho I > 0$. In a practical view, this theory is commonly used for calculations of vibrations of 'sandwich beams', which define beams that have an outer layer of a material with a high axial rigidity (i.e. glass fibre reinforced synthetic material) and inner layers of a softer material. Besides the vertical bending deformation, which is already taken into account for the Euler-Bernoulli beam, also the shear deformation is to be considered if respecting the Timoshenko beam theory. That means, two independent kinds of deformations w and ψ are allowed. Whereas the Euler-Bernoulli beam theory considers the vertical deformation w and the rotation ψ as dependent on each other since it holds $w' = -\psi$.



Bending deformation
(flexion ψ)



shear deformation
(shear distortion γ)

Figure 2.3.: The bending and shear deformation of a Timoshenko beam.

If we take a look back on the earlier derived differential equation (2.9) of a beam, it is known that the equation of motion for a beam with constant cross-section is

$$EIw^{IV} + \rho A \ddot{w} - \rho I \left(1 + \frac{EA}{GA_s}\right) \ddot{w}'' + \rho A \frac{\rho I}{GA_s} \ddot{w} = q + \frac{\rho I}{GA_s} \ddot{q} - \frac{EI}{GA_s} q''. \quad (2.23)$$

The equation (2.23) can be simplified for the case of free vibrations with the condition $q = 0 \rightarrow \ddot{q} = q'' = 0$ to end up with the equation of motion for free vibrations

$$EI \frac{\partial^4 w}{\partial x^4} + \rho A \frac{\partial^2 w}{\partial t^2} - \rho I \left(1 + \frac{EA}{GA_s}\right) \frac{\partial^4 w}{\partial x^2 \partial t^2} + \rho A \frac{\rho I}{GA_s} \frac{\partial^4 w}{\partial t^4} = 0. \quad (2.24)$$

With the Ansatz for harmonic vibrations

$$w_{(x,t)} = W_{(x)} \cos(\omega t - \varphi), \quad (2.25)$$

and the abbreviations

$$\kappa^4 = \omega^2 \frac{\rho A}{E I} [m^{-4}], \quad i^2 = \frac{I}{A} [m^2], \quad \alpha = \frac{E A}{G A_s} [-], \quad (2.26)$$

(where i is the radius of inertia), the general differential equation is obtained as

$$\frac{d^4 W}{dx^4} + \kappa^4 i^2 (1 + \alpha) \frac{d^2 W}{dx^2} - \kappa^4 (1 - \kappa^4 i^4 \alpha) W = 0. \quad (2.27)$$

When the exponential Ansatz for the complex deformation in the frequency domain

$$W_{(x)} = e^{\lambda x}, \quad (2.28)$$

is applied, the equation (2.27) can be expressed as the characteristic polynomial

$$\lambda^4 + \kappa^4 i^2 (1 + \alpha) \lambda^2 - \kappa^4 (1 - \kappa^4 i^4 \alpha) = 0. \quad (2.29)$$

With $\bar{\lambda} = \lambda^2$ and the abbreviations $P = \kappa^4 i^2 (1 + \alpha)$ and $Q = \kappa^4 (1 - \kappa^4 i^4 \alpha)$ the equation (2.29) can be reduced to a second-order eigenvalue problem

$$\bar{\lambda}^2 + P\bar{\lambda} - Q = 0. \quad (2.30)$$

Solving for $\bar{\lambda}$ leads to two solutions

$$\bar{\lambda}_{1,2} = -\frac{P}{2} \pm \sqrt{\left(\frac{P}{2}\right)^2 - Q}. \quad (2.31)$$

Respecting the relation $\lambda = \pm\sqrt{\bar{\lambda}}$ generates two eigenvalues λ for each $\bar{\lambda}$. Thus, the complete set of four eigenvalues is

$$\lambda_1 = (i) \sqrt{-\frac{P}{2} - \sqrt{\left(\frac{P}{2}\right)^2 - Q}}, \quad \lambda_2 = (i) \sqrt{-\frac{P}{2} + \sqrt{\left(\frac{P}{2}\right)^2 - Q}}, \quad (2.32)$$

$$\lambda_3 = -(i) \sqrt{-\frac{P}{2} - \sqrt{\left(\frac{P}{2}\right)^2 - Q}}, \quad \lambda_4 = -(i) \sqrt{-\frac{P}{2} + \sqrt{\left(\frac{P}{2}\right)^2 - Q}}. \quad (2.33)$$

One can observe that each of the four eigenvalues can be, depending on the frequency range, either real or imaginary as a term under any of the two square roots turns negative at a certain value for ω . This specific frequency ω^c , where the eigenvalues change domains is defined as the critical frequency [26]

$$\omega^c = \sqrt{\frac{GA_s}{\rho I}}. \quad (2.34)$$

Signifying that a beam has a critical frequency ω^c depending on its material properties and its eigenvalues change domains within the three defined frequency ranges

$$\begin{aligned} \text{I:} & \quad 0 < \omega < \omega^c, \\ \text{II:} & \quad \omega = \omega^c, \\ \text{III:} & \quad \omega > \omega^c. \end{aligned}$$

Due to this change, different general solutions hold within those intervals

$$\text{I:} \quad W_{(x)} = C_1 \cos(\lambda_1 x) + C_2 \sin(\lambda_1 x) + C_3 e^{\lambda_1(x-L)} + C_4 e^{-\lambda_1 x} \quad \text{for } \omega < \omega^c$$

$$\text{II:} \quad W_{(x)} = C_1 e^{\lambda_1 x} + C_2 e^{\lambda_2 x} + C_3 e^{\lambda_3 x} + C_4 e^{\lambda_4 x} \quad \text{for } \omega = \omega^c$$

$$\text{III:} \quad W_{(x)} = C_1 \cos(\lambda_1 x) + C_2 \sin(\lambda_1 x) + C_3 \cos(\lambda_2 x) + C_4 \sin(\lambda_2 x) \quad \text{for } \omega > \omega^c$$

More details about the behaviour of the eigenvalues can be studied in Appendix B. Besides the transversal deflection, also the cross-sectional rotation of the beam needs to be considered in the Timoshenko beam theory, therefore it holds for ψ considered in the frequency domain

$$\psi_{(x)} = -\frac{\frac{\partial^3 W}{\partial x^3} + \frac{\partial W}{\partial x} \left(\frac{\omega^2 \rho A}{GA_s} + \frac{A_s G}{EI} \right)}{\frac{\omega^2 \rho}{E} - \frac{A_s G}{EI}}. \quad (2.35)$$

The internal forces and the bending moment in the frequency domain for the Timoshenko beam are defined as

$$M_{(x)} = \frac{EI \partial \psi}{\partial x}, \quad V_{(x)} = GA_s \left(\frac{\partial W}{\partial x} - \psi \right) \quad (2.36)$$

2.2. Longitudinal Vibrations

Longitudinal vibrations include dynamic deformation along the longitudinal axis of a beam. It should be mentioned that the system will be acting under free vibrations, meaning that there is a starting deflection, after releasing the deflection the system is moving completely on its own. If a homogenous beam with a constant material density ρ and cross-sectional area A over the length of the beam is considered, the differential equation of longitudinal motion can be obtained through the equilibrium conditions, see Figure 2.4 and (2.37),

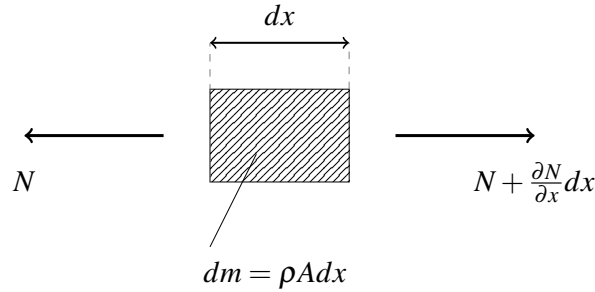


Figure 2.4.: The equilibrium condition for longitudinal forces on a small element.

$$\rho A dx \ddot{u} = -N + \left(N + \frac{\partial N}{\partial x} dx \right) \rightarrow \rho A \ddot{u} = N'. \quad (2.37)$$

When applying the elasticity equation

$$N = E A u' \quad (2.38)$$

one can obtain $\rho \ddot{u} = E u''$ or

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \quad \text{with} \quad c^2 = \frac{E}{\rho}, \quad (2.39)$$

where c defines the wave propagation velocity which depends in case of longitudinal vibrations only on the Young's Modulus E and the material density ρ .

To find a solution for the differential equation, an Ansatz is needed. As we focus on vibrations of finite beams, the Bernoulli Ansatz has been chosen so that

$$u(x, t) = U_{(x)} \cos(\omega t - \varphi), \quad (2.40)$$

whereas a harmonic vibrational behaviour of the beam is assumed. The equation (2.40) can now be applied on the general differential equation (2.39) to obtain

$$\frac{d^2 U}{dx^2} + \left(\frac{\omega}{c}\right)^2 U = 0 \quad (2.41)$$

with the general solution in the frequency domain

$$U_{(x)} = C_5 \cos\left(\frac{\omega}{c}x\right) + C_6 \sin\left(\frac{\omega}{c}x\right). \quad (2.42)$$

Applying the equation (2.42) into the Ansatz (2.40) leads to

$$u_{(x)} = \left[C_5 \cos\left(\frac{\omega}{c}x\right) + C_6 \sin\left(\frac{\omega}{c}x\right) \right] \cos(\omega t - \varphi), \quad (2.43)$$

being the general solution in the time domain.

3. APPLICATION

In this chapter, the derived differential equations are applied to structures. The calculation steps to obtain the natural frequencies and mode shapes of structures are described generally. The computational implementation of those steps explained in Chapter 3 are then described in Chapter 6.

3.1. Structural Definitions

It is of importance to define the frame structure systematically [27]. Therefore, the wanted object can be seen as a set of frame members τ and frame nodes N_τ . Each frame node is either considered as an unsupported node $N_F \subsetneq N_\tau$ or a supported node $N_S \subsetneq N_\tau$, but can not be both. Thus, for the frame nodes of the structure it holds that $N_\tau = N_F + N_S$. Each frame member $m \in \tau$ connects in each case two frame nodes $k \in N_\tau$ which are denoted as $\{k_{m0}, k_{mL}\} =: \varkappa_m \subset N_\tau$. The parameters L_m, E_m, G_m, A_m and I_m can be defined for each frame member individually. For practical reasons the unit tangential vector $\{r\}_m$ is established for each member m pointing from k_{m0} to k_{mL} , along with the corresponding unit normal vectors. Furthermore, each frame node $k \in N_\tau$ has a set of attached frame members $\beta_k \subset \tau$. If a change in parameters is wished within a frame member, two frame members need to be modelled as one can only have constant parameters. Any number of bearings and springs can be applied in a combinational matter on then called, supported nodes N_S .

3.2. Differential Equations

For each frame member $m \in \tau$ the general solution of the differential equations for longitudinal and transversal vibrations in the frequency domain U_m and W_m respectively are set up. To enable the enforcement of boundary and intersectional conditions the kinematic relations N_m, V_m and M_m are defined based on the derivatives of the general solutions U'_m, W'_m, W''_m and W'''_m with respect to the local beam axis x . The setup of kinematic and kinetic equations differ depending on the chosen beam theory.

The explanation is based on vectors and matrices that are denoted using $\{\}$ and $[\]$ parenthesis, respectively. To start the application the resulting internal force S_k at a node k is introduced by adding the vectors containing the kinetic forces in the local coordinate system through (3.1)

$$\{S_k\} = \{N_k\} + \{V_k\}. \quad (3.1)$$

Same holds true for the total deflection $\{\bar{U}_k\}$ of a frame member at this node that can be expressed as the combination of the local transversal and longitudinal displacement $\{W_k\}$ and $\{U_k\}$ respectively, see Figure 3.1 and (3.2)

$$\{\bar{U}_k\} = \{U_k\} + \{W_k\}. \quad (3.2)$$

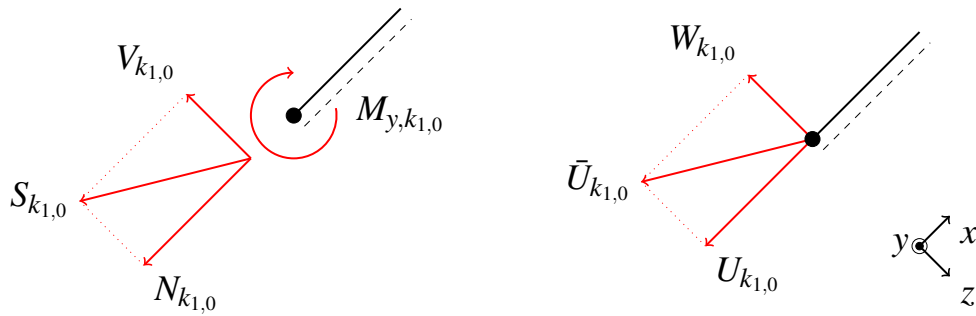


Figure 3.1.: The resulting kinetic force S (left) and the total deformation \bar{U} (right) at the frame node $k_{1,0}$.

The kinetic force S_k for each frame member $m \in \tau$ is globally composed by a horizontal and vertical component in the global coordinate system as $F_{X,k}$ and $F_{Z,k}$ respectively. The global coordinate system serves as the basis for attachments of bearings and springs at any angle. Therefore, the decomposition of S_k into $F_{X,k}$ and $F_{Z,k}$ is essential to facilitate the computational steps in Chapter 6. The same holds for the resulting deflection \bar{U} , which can be decomposed into $U_{X,k}$ and $U_{Z,k}$

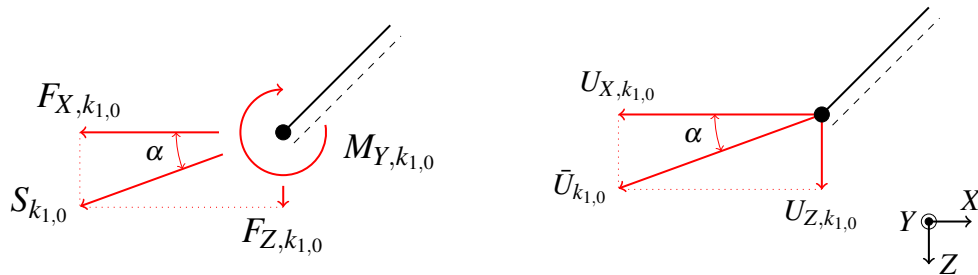


Figure 3.2.: The decomposed kinetic force (left) and the total deformation (right) at the frame node $k_{1,0}$.

3.3. Boundary Conditions

The general equilibrium conditions $\{F_{eq}\} := \sum \{F_k\} = 0$ and $\{M_{eq}\} := \sum M_k = 0$ are formulated for each frame node $k \in N_\tau$ by adding the kinematic quantities $\{F_k\} = \{F_{X,k}, F_{Z,k}\}^T$ and $M_{Y,k}$ of each attached frame member $\beta_k \in \tau$ at the corresponding node k_{m0} or k_{mL} . Resulting in n_N equations, where n_N is the number of frame nodes N_τ . An example can be studied, where the internal forces are exemplified for an arbitrary frame structure in Figure 3.3.

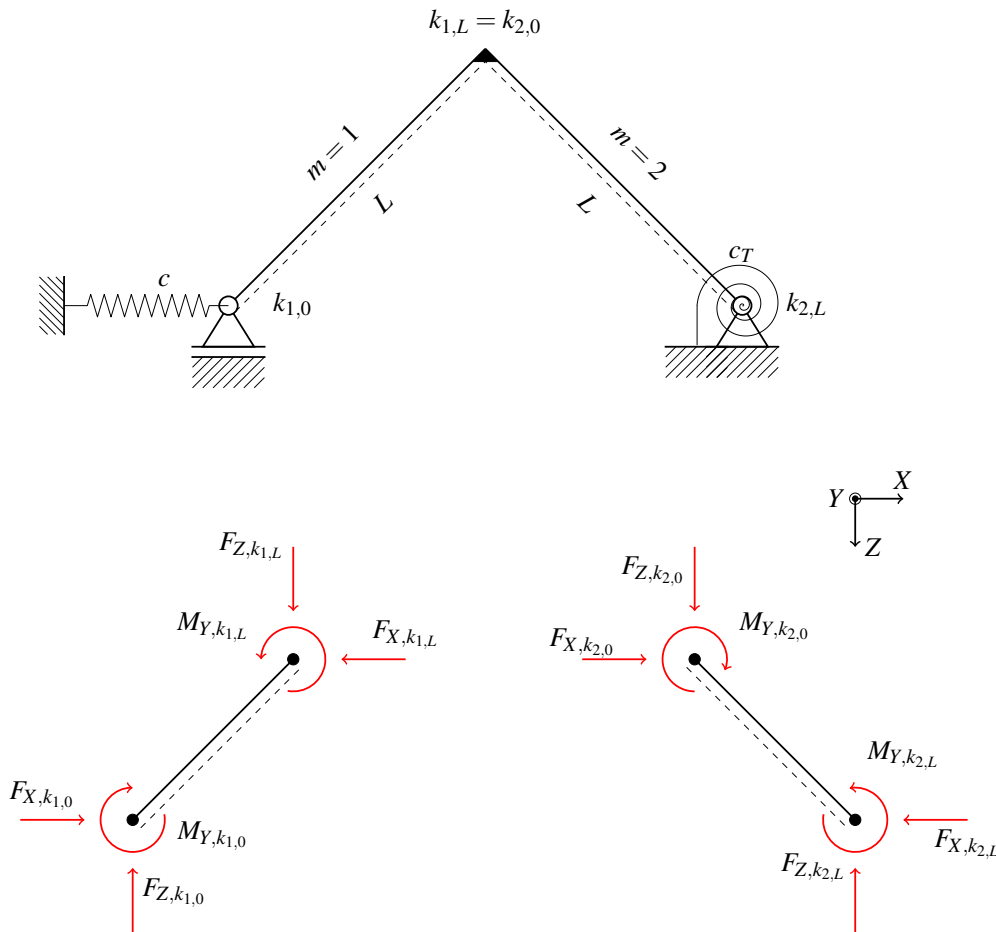


Figure 3.3.: The equilibrium conditions at each frame node $k \in N_\tau$ (bottom), applied at an example frame (top).

A collection of equations $\{B_{eq}\} = 0$ is defined by setting the constrained kinematic or kinetic quantities due to attached bearings zero at the corresponding supported frame node

$k \in N_S$. Simultaneously, the a variable is introduced representing the unknown kinematic and/or kinetic quantities at this node. All unknown quantities are stored in a separate vector $\{Q\}$ and will be determined along with the integration constants $\{C\}_{1-6}$ when solving the system equation. In the case of a roller, this would mean that the deformation perpendicular to the rolling surface and the rotational moment at this node equals zero, see Figure 3.4 and the kinetic variable $\{F_n\}$ is added to $\{Q\}$. Since bearings can be attached in any slanted position, the perpendicular kinematic and kinetic components U_n and F_n , as well as the parallel components U_p and F_p respectively, are decomposed into quantities $\{U_X, U_Z\}^T$ and $\{F_X, F_Z\}^T$ respecting the global coordinate system. Further bearing conditions can be obtained from Table 3.1.

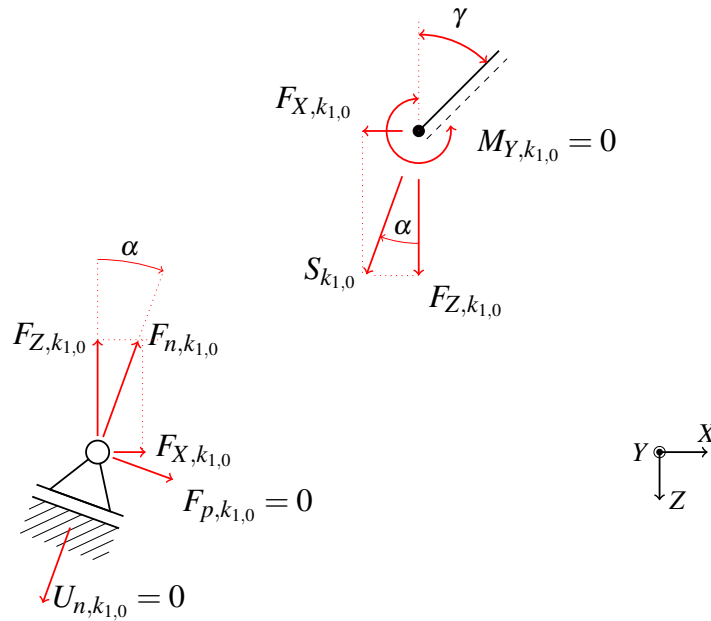


Figure 3.4.: Attachment of a roller in a slanted position.

Respecting longitudinal and rotational springs, the corresponding equilibrium equation at this node $\{F_k\} = 0$ and $M_k = 0$ is modified by adding the increase of the internal force and bending moment respectively due to the spring equations

$$\Delta F = c \cdot \Delta U \quad \text{and} \quad \Delta M = c \cdot \Delta \psi. \quad (3.3)$$

For longitudinal springs, the translational spring force is decomposed into the X and Z components regarding the global coordinate system and added as ΔF_X and ΔF_Z respectively as the spring can be applied at an angle α .

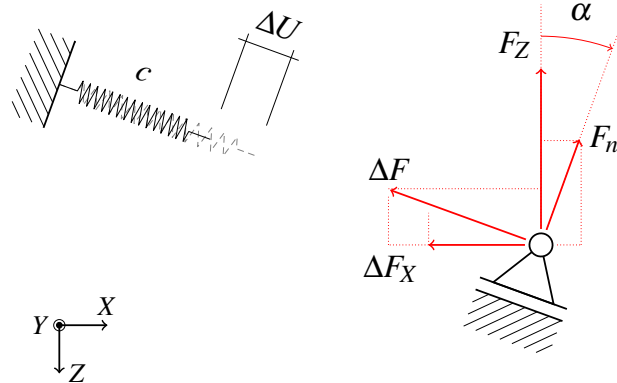


Figure 3.5.: Attachment of a longitudinal spring.

While the geometrical components play an important role when applying translational springs, the rotational spring can be attached in any case when referring to the two-dimensional space. Meaning, an additional rotational moment around the global Y -axis is added to the equilibrium equation. It holds $M_k = \Delta M$ at the supported node $k \in N_S$, as seen in Figure 3.6. Further spring conditions can be obtained from Table 3.2.

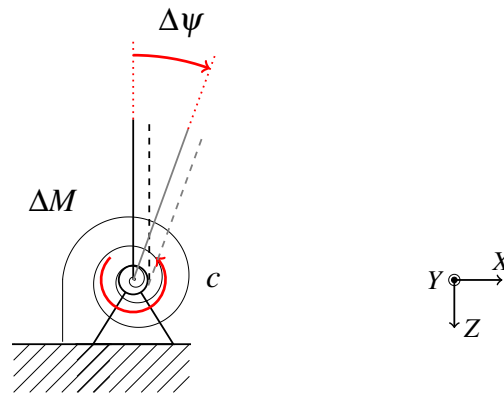


Figure 3.6.: Attachment of a rotational spring.

Thus far, the explained bearing and spring conditions hold for all supported nodes $k \in N_S$. Kinetic quantities such as $\{F_k\}$ and M_k remain zero at nodes without attached bearings or springs noted as $k \in N_F$.

An additional set $\{J_{eq}\} = \{0\}$ of coupling conditions need to be defined for nodes with a set of attached frame members β_k where $\#\{m \in \beta_k : m > 1\}$. To set up the correct kinematic

and kinetic relations among the attached frame members we introduce one fixed frame member $m_f \in \beta_k$. The kinematic compatibility equations can now be defined for any case as

$$\bar{U}_{m_f|k} = \bar{U}_{n|k} \quad \forall n \in \beta_k \setminus m_f. \quad (3.4)$$

Only when connecting beams rigidly, the rotation angle of this node is





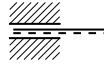
$$\bar{\Psi}_{m_f|k} = \bar{\Psi}_{n|k} \quad \forall n \in \beta_k \setminus m_f. \quad (3.5)$$

Contrarily, for beams connected with hinges the condition for the rotational moment is

$$M_n = 0 \quad \forall n \in \beta_k. \quad (3.6)$$

A vector $\{eq\}$ is proposed and can be seen as the collection of system equations $\{F_{eq}, M_{eq}, B_{eq}, J_{eq}\}^T$. Consequently, the system equation is of the form $\{eq\} = \{0\}$ where $\{eq\}$ depends on the integration constants C_1, \dots, C_6 , the set of bearing constrains Q and ω .

Table 3.1.: A list of the different support types and its support conditions.

name of support type	symbolic representation	supporting conditions
Pinned		$M = 0$ $U_x = 0$ $U_z = 0$
Roller		$M = 0$ $S_x = 0$ $U_z = 0$
Fixed		$\Psi = 0$ $U_x = 0$ $U_z = 0$
Parallel guide		$S_z = 0$ $U_x = 0$ $\Psi = 0$
Sliding Sleeve		$S_x = 0$ $U_z = 0$ $\Psi = 0$

where

$$\begin{aligned}
 B &= EI & P &= AE & a &= B \omega \beta^2 & c &= P \omega \alpha \\
 \alpha &= \sqrt{\frac{\rho}{E}} & \gamma &= L \sqrt{\omega} \beta & \text{and} & b &= B \omega^{3/2} \beta^3 & d &= \sqrt{\omega} \beta \\
 \beta &= \left(\frac{A \rho}{EI}\right)^{1/4} & \delta &= L \omega \alpha
 \end{aligned}$$

Non trivial solutions for ω fulfil the system of equation (3.7) when its determinant is zero,

$$f(\omega) = \det(\mathbf{e}\mathbf{q}) = 0. \quad (3.8)$$

$f(\omega)$ is a continuous function with an infinite number of zero crossings and is often referred to as the eigen-, or frequency function, or characteristic equation.

In the case of an Euler-Bernoulli cantilever beam, the eigenfunction is

$$f(\omega) = 2PB^2 \omega^4 \rho^2 \alpha \beta^6 \cos(\delta) e^{-2\gamma} [\cos(\gamma) + 2e^\gamma + e^{2\gamma} \cos(\gamma)]. \quad (3.9)$$

The frequency function of a cantilever beam is plotted along the ω axis and can be studied in Figure 3.7. Certainly, the shape of $f(\omega)$ depends on the structural parameters, and so do the natural frequencies.

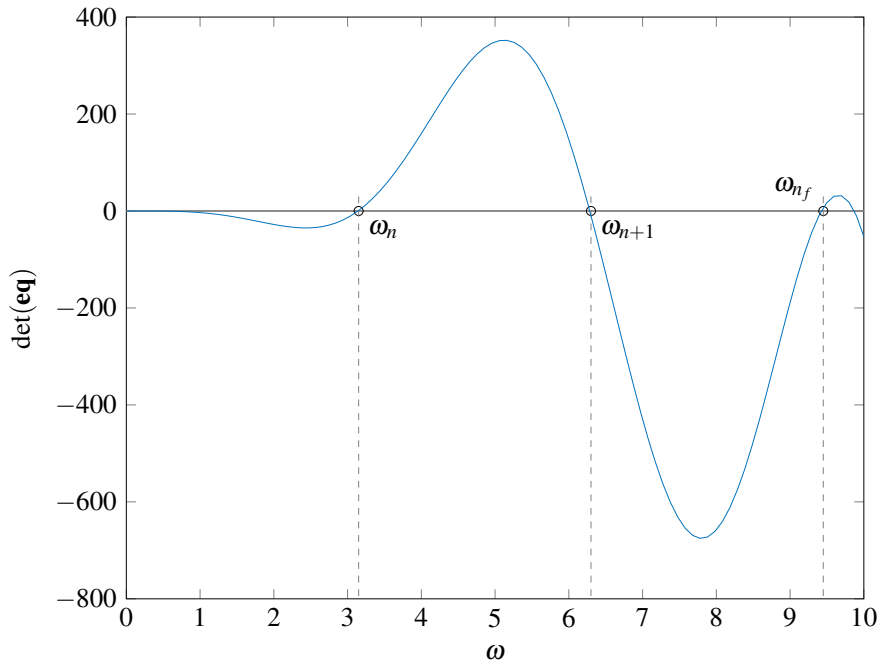


Figure 3.7.: Plot of the characteristic function for a Euler-Bernoulli cantilever beam in the range of $\omega = 0 - 10 \text{ rad/s}$.

3.5. Natural Frequencies

Each ω that fulfils the so-called characteristic equation $f(\omega) \stackrel{!}{=} 0$ is an eigenvalue of the system of equations and therefore a natural frequency of the frame structure. Theoretically, there is an infinite number of eigenfrequencies for any structure since the frequency function is continuous and provides an endless number of zero crossings.

However, in finite element approaches, the number of natural frequencies is limited to the number of degrees of freedom and the determination of the natural frequencies ω is rather convenient, since the equation of an undamped motion without an excitation force is

$$\mathbf{M}\ddot{w}_{(x,t)} + \mathbf{K}w_{(x,t)} = 0. \quad (3.10)$$

with the assumption of harmonic vibrations $w_{(x,t)} = W_{(x)} \cos(\omega t - \varphi)$ the equation (3.10) can be written in the frequency domain as

$$\mathbf{K} - \omega^2 \mathbf{M} = 0. \quad (3.11)$$

To determine the natural frequencies, the non-trivial solutions for ω need to be found by setting the determinant of the system equation zero,

$$f(\omega) = \det(\mathbf{K} - \omega^2\mathbf{M}) = 0. \quad (3.12)$$

Finding ω 's that full fill (3.12) requires less computational effort since ω is represented explicitly in the system equation. Whereas in the formulation using the Numerical Assembly Technique, the variable ω is implicitly included in the system matrix [eq]. Hence, finding the zeros of the frequency function $f(\omega) = \det(\mathbf{eq})$ can be rather laborious.

Determination of zeros

The zeros of $f(\omega)$ are determined using the Regular-Falsi method. Evaluating $f(\omega)$ step by step at a_k and b_k using rather big intervals, sign changes can be spotted which locates the area of a potential zero-crossing roughly. When a sign change is found, the exact value for $f(\omega) = 0$ is obtained by evaluating the function value at the intersection point S using progressively decreasing interval sizes, within this determined area. S can be calculated through

$$S = \frac{a_{k-1}f(b_{k-1}) - b_{k-1}f(a_{k-1})}{f(b_{k-1}) - f(a_{k-1})}, \quad (3.13)$$

and is therefore, the intersection point of the secant going through the points (a_k, f_{a_k}) and (b_k, f_{b_k}) , and the ω -axis. As soon as the function value $f_{c_{k+1}}$ at a specific intersection point $S^* = c_{k+1}$ is zero, thus, $f(S^*) = 0$, a natural frequency $\omega = S^*$ is found. An illustration of one iteration step can be seen in Figure 3.8. This method gives accurate results with a rather short computation time and, as a result, meets the expectations of this application.

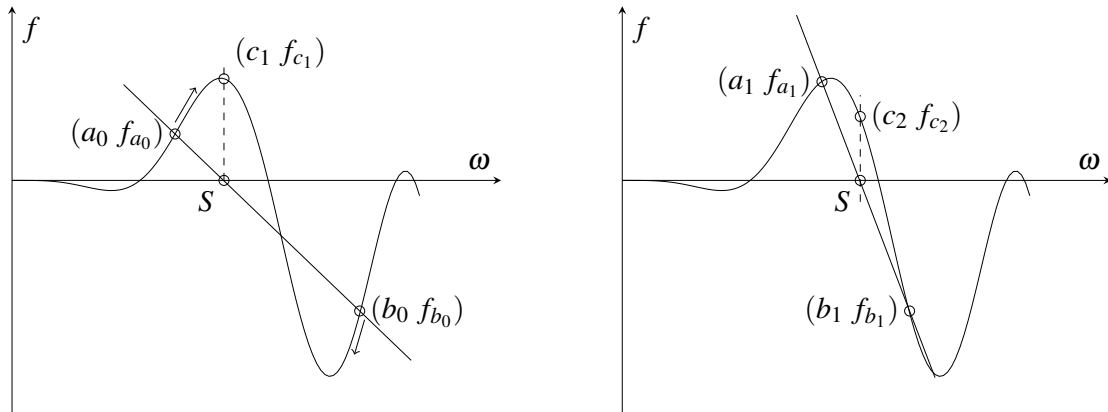


Figure 3.8.: One iteration-step of the regula-falsi method.

Collection of characteristic functions

To validate the computations, the characteristic equation can be solved manually. Therefore, lists of characteristic functions and mode shapes for transversal Table 3.3 and longitudinal vibrations Table 3.4 of some common structures are added where $k_B = \sqrt{EI/\rho AL^4}$ and $k_D = \sqrt{E/\rho L^2}$ [2]. It should be noted that the characteristic functions shown in Table 3.3 and Table 3.4 are considering transversal and longitudinal vibrations separately. As soon as both vibrations are simultaneously considered in the calculation, the characteristic equations become more complex. In this thesis, the transversal and longitudinal vibrations are taken into account. Therefore different characteristic functions will be shown in Chapter 4, where the NAT will be executed on test examples.

Further note, the mode shapes for longitudinal vibrations in Table 3.4 describe the distribution of the axial deflection along the beam.

Table 3.3.: Frequency functions for transversal vibrations.

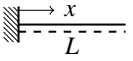
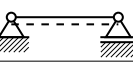
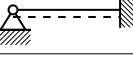
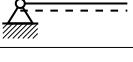
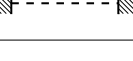

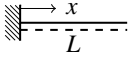
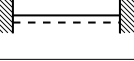
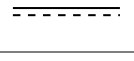
System: $\xi = \frac{x}{L}$	frequency function	$\omega_j = \lambda_j^2 \cdot k_B$	mode shape
	$\cos \lambda \cosh \lambda = -1$	$\lambda_j = (j - \frac{1}{2})\pi$	$\frac{\sin \lambda \xi - \sinh \lambda \xi}{\sin \lambda + \sinh \lambda} - \frac{\cos \lambda \xi - \cosh \lambda \xi}{\cos \lambda + \cosh \lambda}$
	$\sin \lambda = 0$	$\lambda_j = j \cdot \pi$	$\sin \lambda \xi$
	$\tanh \lambda = \tan \lambda$	$\lambda_j = (j + \frac{1}{4})\pi$	$\frac{\sin \lambda \xi}{\sin \lambda} - \frac{\sinh \lambda \xi}{\sinh \lambda}$
	$\tanh \lambda = \tan \lambda$	$\lambda_j = (j + \frac{1}{4})\pi$	$\frac{\sin \lambda \xi}{\sin \lambda} + \frac{\sinh \lambda \xi}{\sinh \lambda}$
	$\cosh \lambda \cos \lambda = 1$	$\lambda_j = (j + \frac{1}{2})\pi$	$\frac{\sin \lambda \xi - \sinh \lambda \xi}{\sin \lambda - \sinh \lambda} + \frac{\cos \lambda \xi - \cosh \lambda \xi}{\cos \lambda - \cosh \lambda}$
	$\cosh \lambda \cos \lambda = 1$	$\lambda_j = (j + \frac{1}{2})\pi$	$\frac{\sin \lambda \xi + \sinh \lambda \xi}{\sin \lambda - \sinh \lambda} - \frac{\cos \lambda \xi + \cosh \lambda \xi}{\cos \lambda - \cosh \lambda}$

Table 3.4.: Frequency functions for longitudinal vibrations.

System: $\xi = \frac{x}{L}$	frequency function	$\omega_j = \lambda_j^2 \cdot k_D$	mode shape
	$\cos \lambda = 0$	$\lambda_j = (j - \frac{1}{2})\pi$	$\sin \lambda \xi$
	$\sin \lambda = 0$	$\lambda_j = j \cdot \pi$	$\sin \lambda \xi$
	$\sin \lambda = 0$	$\lambda_j = j \cdot \pi$	$\cos \lambda \xi$

3.6. Mode Shapes

Implementing the calculated ω -values into our system of equation makes it possible to solve for the integration constants $\{C_1, \dots, C_6\}$ and the bearing constrains $\{Q\}$ stored in $\{\bar{C}\}$. Considering that we have n - number of natural frequencies we also have n - corresponding system matrices. Through determining the eigenvalues and eigenvectors of each system matrix, n - matrices containing the eigenvectors and n - diagonal matrices representing the eigenvalues are obtained.

To get the corresponding mode shapes to each natural frequency, the column of the matrix containing the eigenvectors needs to be found where the associated eigenvalue reaches the minimum value. A brief example is given where $[\mathbf{v}]$ is the matrix of eigenvectors and $[\lambda]$ the diagonal matrix of eigenvalues.

$$\mathbf{v}_i = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 & 25 \end{bmatrix} \quad \lambda_i = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 \\ & 7 & 0 & 0 & 0 \\ & & \text{sym.} & 19 & 0 \\ & & & & 21 \end{bmatrix}$$

Figure 3.9.: The matrix of eigenvectors $[\mathbf{v}_i]$ and eigenvalues $[\lambda_i]$ for one specific natural frequency ω_i .

A more practical example is shown where the matrix of eigenvectors and eigenvalues are given for the first transversal mode shape of a cantilever beam. In this case, the seventh column of $[\lambda_1]$ has the lowest eigenvalue on the main diagonal and is, therefore, the searched column number in $[\mathbf{v}_1]$.

$$\lambda_1 = \begin{bmatrix} 12.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & 8.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & 2.6 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & 0.27 & 0 & 0 & 0 & 0 & 0 \\ & & & & 49.0 & 0 & 0 & 0 & 0 \\ & & \text{sym.} & & & 49.0 & 0 & 0 & 0 \\ & & & & & & 0 & 0 & 0 \\ & & & & & & & 0.08 & 0 \\ & & & & & & & & 4.6 \end{bmatrix}$$

Figure 3.10.: The matrix of eigenvalues for a cantilever beam evaluated at the first eigenfrequency ω_1 , highlighting the j^{th} column containing the lowest eigenvalue on the main diagonal.

$$\mathbf{v}_1 = \begin{bmatrix} 3 \cdot 10^{-4} & 0.31 & 0.78 & 0.2 & 0.05 & 0.05 & 0.083 & 6 \cdot 10^{-5} & 0.13 \\ 3 \cdot 10^{-4} & 0.08 & 0.23 & 0.1 & 0.17 & 0.17 & 0.061 & 2 \cdot 10^{-4} & 0.65 \\ 8 \cdot 10^{-4} & 0.4 & 0.09 & 0.04 & 0.68 & 0.68 & 0.083 & 3 \cdot 10^{-4} & 0.43 \\ 4 \cdot 10^{-4} & 0.31 & 0.29 & 0.01 & 0.02 & 0.02 & 0.061 & 2 \cdot 10^{-4} & 0.4 \\ 1.0 & 0.79 & 0.02 & 7 \cdot 10^{-4} & 0.65 & 0.65 & 0 & 0.08 & 0.12 \\ 6 \cdot 10^{-4} & 0.03 & 2 \cdot 10^{-3} & 7 \cdot 10^{-5} & 0.3 & 0.3 & 0 & 9 \cdot 10^{-3} & 0.02 \\ 0.08 & 0.1 & 9 \cdot 10^{-3} & 2 \cdot 10^{-3} & 0.01 & 0.01 & 0 & 1.0 & 0.03 \\ 4 \cdot 10^{-5} & 0.01 & 0.33 & 0.59 & 0.01 & 0.01 & 0.8 & 3 \cdot 10^{-3} & 0.12 \\ 8 \cdot 10^{-6} & 0.05 & 0.37 & 0.77 & 7 \cdot 10^{-3} & 7 \cdot 10^{-3} & 0.58 & 1 \cdot 10^{-3} & 0.43 \end{bmatrix}$$

Figure 3.11.: The matrix of eigenvectors for a cantilever beam evaluated at the first eigenfrequency ω_1 , highlighting the j^{th} column responsible for the first transversal eigenmode.

Hence, the first six entries of the vector containing the eigenmode are the values for the integration constants $\{C_1, \dots, C_6\}$ and the last three entries are related to the restricted degrees of freedom through enforcement of the boundary conditions. It can be noted that for the first transversal mode shape the integration constants $\{C_5, C_6\}$ remain zero as longitudinal deformation does not contribute to this mode shape. Also the X component of the supporting force F_X is zero whereas the Z component F_Z and the bending moment M_Y do interact. Thus, all variables are determined to get the transversal deformation $W_{(x)}$ or so-called mode shapes for each frame member and natural frequency. The contribution of the integration constants and bearing variables stored in $\{\bar{C}\}$ to the first transversal eigenmode and a visualization of the the mode shape can be seen in Figure 3.12 and Figure 3.13 respectively.

$$\begin{bmatrix} 0.083 \\ 0.061 \\ 0.083 \\ 0.061 \\ 0 \\ 0 \\ 0 \\ 0.8 \\ 0.58 \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ F_X \\ F_Z \\ M_Y \end{bmatrix} \rightarrow \begin{aligned} W_{(x)} &= C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 e^{\kappa(x-L)} + C_4 e^{-\kappa x} \\ U_{(x)} &= C_5 \cos \frac{\omega}{c} x + C_6 \sin \frac{\omega}{c} x = 0 \end{aligned}$$

Figure 3.12.: The contribution of the integration constants and bearing variables on the first transversal mode shape of a cantilever beam.

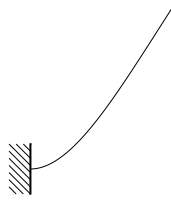


Figure 3.13.: The first transversal mode shape of a cantilever beam.

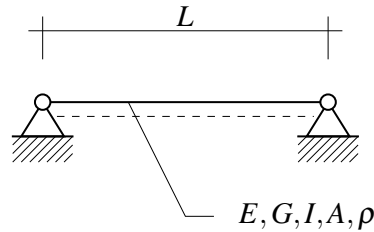
4. TEST EXAMPLES

The calculation process was applied to different structural examples such as a single-span beam (pinned-pinned), a clamped-pinned beam, a cantilever beam (clamped-free), and an arbitrary structure consisting of two rigidly connected beams. Those examples enable a comparison of the natural frequencies calculated based on the two different beam theories used. General parameters were chosen to be able to retrace the calculation process better and to get a good variety of transversal and longitudinal vibrations in the results. To study the difference between the Euler-Bernoulli and the Timoshenko beam theory, the beams were kept compact, meaning the rate L/I is rather small in comparison to conventional beams. The code-based computations were double-checked, by comparing the computed solutions for the eigenfrequencies to hand calculations. Especially for the more simple structures, this was done by using existing analytic formulas. For unconventional structures, the results were checked through finite element-based computations using at least ten elements per beam.

4.1. Single-Span Beam

The single-span beam is the most common example structure, to begin with, and was executed under the following general parameters

$$\begin{aligned} E &= 1 && [kN/m^2] \\ G &= 1 && [kN/m^2] \\ I &= 1 && [m^4] \\ A &= 1 \cdot 10^3 && [m^2] \\ L &= 1 && [m] \\ \rho &= 1 && [t/m^3]. \end{aligned}$$



The transversal natural frequencies were calculated by hand using analytically derived formulas [4] for the Euler-Bernoulli and Timoshenko beam theory. Furthermore, the structure was analysed using the computational code to determine the eigenfrequencies.

Hand Calculation

The transversal eigenfrequencies of a single-span beam based on the Euler-Bernoulli beam theory can be obtained through the equation

$$\omega_{B,n}^2 = n_f^4 \omega_B^2 \quad (4.1)$$

with ω_B as the natural frequency due to bending deformation and n_f — the number of the wished natural frequency. ω_B depends on the bending stiffness B , the mass distribution μ and the length L of the beam. Thus, the natural frequencies can be calculated through

$$\omega_B^2 = \frac{B\pi^4}{\mu L^4} \quad (4.2) \quad \begin{array}{l} \text{with:} \\ B = EI \\ \mu = \rho A \end{array}$$

The results for the first four natural frequencies were added in Table 4.1 to compare them to the computations.

To calculate the natural frequencies based on the Timoshenko beam theory by hand, the shear stiffness $S = GA_s$ is required. Since the deformation of a Timoshenko beam is a composition of bending and shear deformation, also the natural frequency $\omega_{BS,n}$ consists of the previously calculated natural frequency due to bending ω_B respecting the Euler-Bernoulli beam theory and an additional part ω_S due to shear deformation respecting the Timoshenko beam theory and is combined as followed

$$\frac{1}{\omega_{BS,n_f}^2} = \frac{1}{n_f^4 \omega_B^2} + \frac{1}{n_f^2 \omega_S^2} \quad (4.3) \quad \begin{array}{l} \text{with:} \\ \omega_S^2 = \frac{S\pi^2}{\mu I^2} \\ S = GA_s \\ A_s = 5/6 A \end{array}$$

The effective shear area A_s was chosen based on a rectangular cross-section and is, therefore, 5/6 of the actual cross-section A .

To calculate the first longitudinal eigenfrequency, one can refer to the general solution

$$U_{(x)} = C_1 \cos\left(\frac{\omega}{c}x\right) + C_2 \sin\left(\frac{\omega}{c}x\right). \quad (4.4)$$

Embedding the boundary conditions

$$\begin{array}{l} U_{(x=0)} = 0 \quad \rightarrow \quad C_1 = 0, \\ U_{(x=L)} = 0 \quad \rightarrow \quad C_2 \sin\left(\frac{\omega}{c}L\right) = 0 \end{array}$$

leads to

$$\sin\left(\frac{\omega}{c}L\right) = 0 \quad \rightarrow \quad \frac{\omega}{c}L = n_f \pi, \quad \rightarrow \quad \omega_{n_f} = \frac{c}{L} n_f \pi$$

Meaning, for non-trivial solutions the term within the sinus function needs to be a multiple n_f of π , whereas n_f is a real and positive integer $n_f \in \mathbb{N}$.

Therefore, the first natural longitudinal frequency of the beam is

$$\omega_{n_f} = \frac{\pi c}{L} n_f \rightarrow \omega_1 = \frac{\pi E}{L \rho} = \pi = 3.141593 \text{ rad/s}$$

The results are added to Table 4.1.

Computation using the NAT

The solution procedure was applied to the single-span beam leading to the characteristic equation, here for the Euler-Bernoulli beam theory

$$f(\omega) = 4AEI\omega^2 \rho \sin(\delta) e^{-2\gamma} \sin(\gamma) (e^{2\gamma} - 1) \quad (4.5)$$

where

$$\gamma = L \sqrt[4]{\frac{\omega^2 A \rho}{EI}}$$

$$\delta = L \omega \sqrt{\frac{\rho}{E}}$$

The complexity and length of the characteristic function regarding the Timoshenko beam theory did not allow it to include it in the report, therefore only the Euler-Bernoulli version of the characteristic function is presented. A plot of the characteristic function in the range of $0 - 4[\text{rad/s}]$ can be seen in Figure 4.1. The zero crossings of this function, so that, $f(\omega) \stackrel{!}{=} 0$ form the natural frequencies and can be studied in Table 4.1.

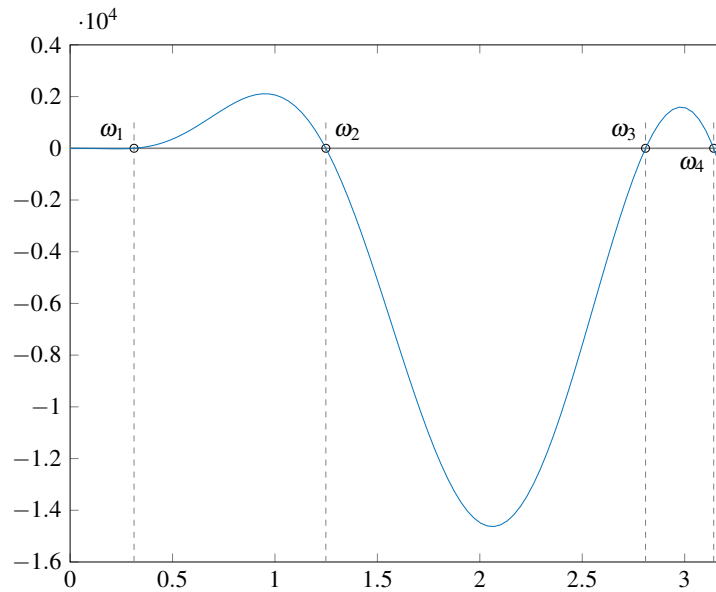


Figure 4.1.: Characteristic function for a single-span beam based on the Euler-Bernoulli beam theory.

Table 4.1.: The natural frequencies of the single-span beam in [rad/s].

no. of eigenmode	Euler-Bernoulli	Timoshenko	hand calculation Euler-Bernoulli	hand calculation Timoshenko
1 (transv.)	0.312104	0.308787	0.312104	0.310272
2 (transv.)	1.248417	1.198448	1.248417	1.219857
3 (transv.)	2.808939	2.577719	2.808939	2.670229
4 (long.)	3.141593	3.141593	3.141593	3.141593

In Table 4.1 it can be seen that the results from the computation are reliable. Also, the natural frequencies respecting the Timoshenko beam theory are slightly lower than the natural frequencies from the Euler-Bernoulli calculations. Thus, the Timoshenko beam behaves less stiff due to the additional shear deformation.

After obtaining the eigenvectors corresponding each natural frequency, the mode shapes were plotted separately in Figure 4.2 and collectively in Figure 4.3. To verify if the kinetic boundary conditions were implemented correctly, the qualitative internal forces occurring at the natural frequencies were observed, see Figure 4.4.

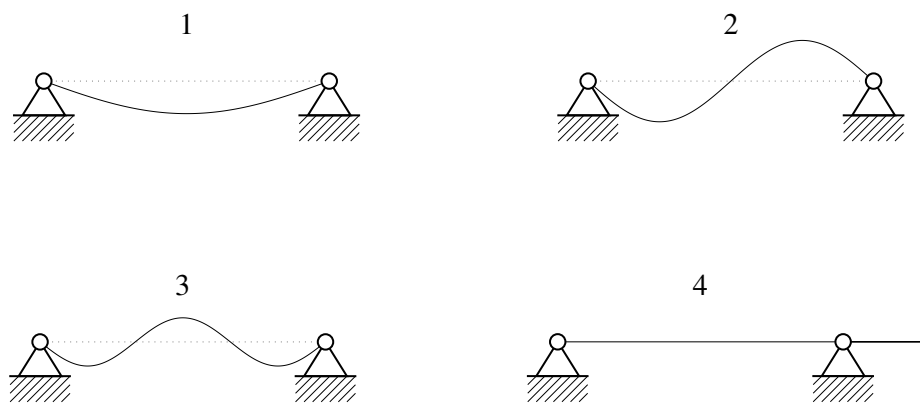
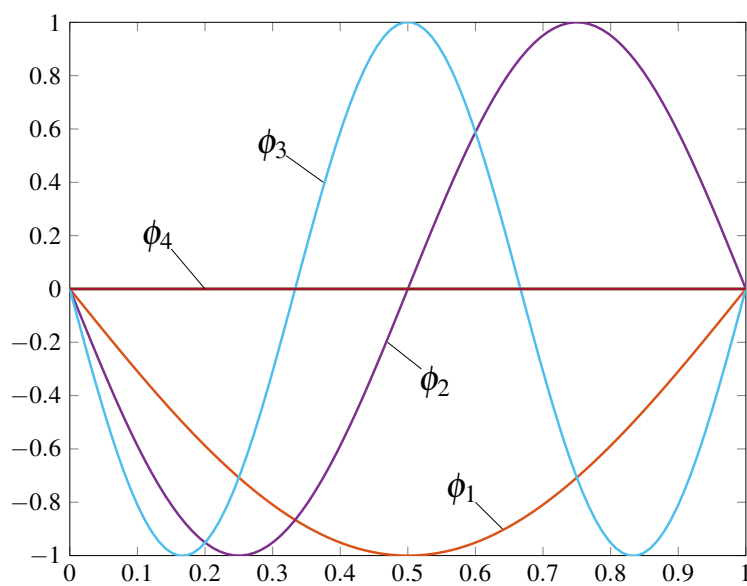


Figure 4.2.: The first four eigenmodes of a single-span beam.

Figure 4.3.: The eigenmodes ϕ_i with $i = \{1, \dots, 4\}$ of a single-span beam in one plot.

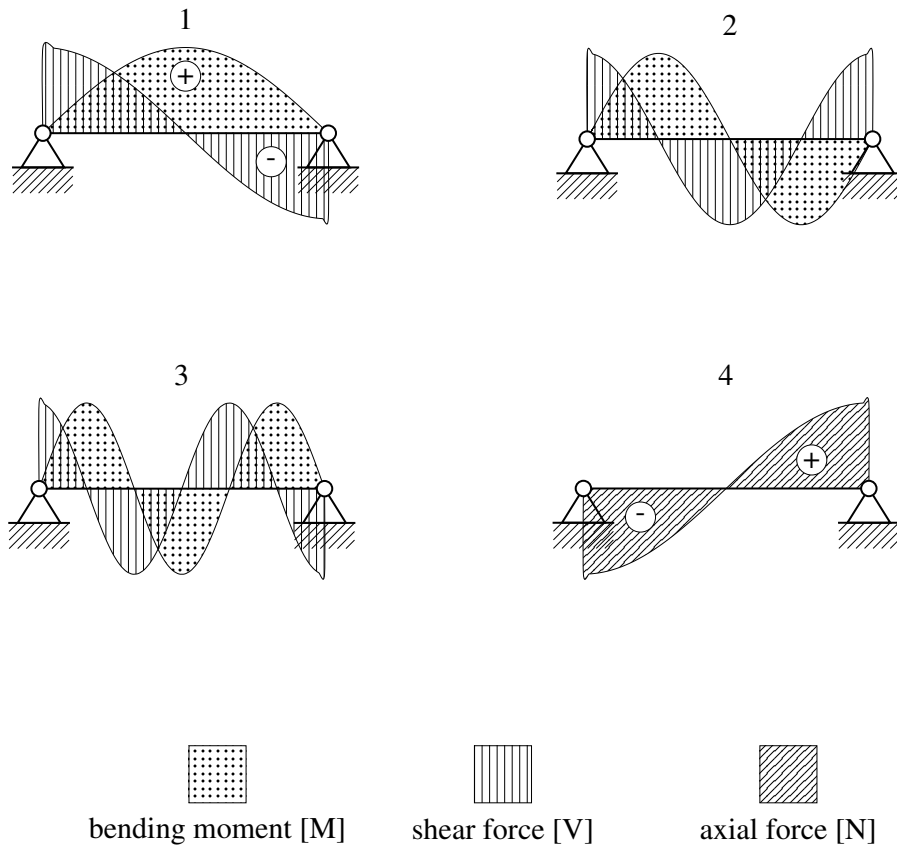
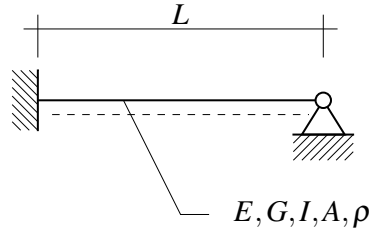


Figure 4.4.: The quantitative internal forces occurring at the natural frequencies 1-4.

4.2. Clamped-Pinned Beam

The single clamped-pinned beam example was executed under the following general parameters

$$\begin{aligned}
 E &= 1 && [kN/m^2] \\
 G &= 1 && [kN/m^2] \\
 I &= 1 && [m^4] \\
 A &= 1 \cdot 10^3 && [m^2] \\
 L &= 1 && [m] \\
 \rho &= 1 && [t/m^3].
 \end{aligned}$$



Hand calculation

To approximately check the reliability of the computation, the first transversal eigenfrequency can be calculated by transforming the beam into a single mass-spring system. The spring stiffness is therefore the force that is needed to enforce a unit displacement at the location where the highest deformation is expected [28].

$$W_{max} = \frac{FL^3}{48\sqrt{5}EI} \rightarrow F_{(W=1)} = k = \frac{48\sqrt{5}EI}{L^3} = 48\sqrt{5} \text{ kN/m} \quad (4.6)$$

The contributing mass m is approximated to be half of the beam's mass. The other half is bared by the supports.

$$m = \rho A \frac{L}{2} = 500 \text{ t}. \quad (4.7)$$

Hence, the first natural frequency can be estimated as

$$\omega = \sqrt{\frac{k}{m}} = \sqrt{\frac{48\sqrt{5}}{500}} = 0.4633 \text{ rad/s}. \quad (4.8)$$

Although this is a fast and simple approximation to check if the computations are reliable, the accuracy of this outcome is rather imprecise. Therefore, the system has been modelled in a finite element-based computation code using ten elements, the results are added in the Table 4.2.

The natural frequencies for longitudinal vibrations remain the same for this example since the boundary conditions in x -direction are the same as for the single-span beam.

Computation using the NAT

The solution procedure was applied to this structure leading to the characteristic function for the Euler-Bernoulli beam

$$f(\omega) = 2EI\omega^{\frac{3}{2}} \sin(\alpha) e^{-2\gamma} [\cos(\gamma) + \sin(\gamma) - e^{2\gamma} \cos(\gamma) + e^{2\gamma} \sin(\gamma)] \delta^{3/4} \quad (4.9)$$

where

$$\alpha = L\omega \sqrt{\frac{\rho}{E}}$$

$$\delta = \frac{A\rho}{EI}$$

$$\gamma = L\sqrt[4]{\omega^2 \delta}.$$

The natural frequencies are the zeros of the function $f(\omega)$. A plot of the frequency function can be seen in Figure 4.5, and the natural frequencies are added in Table 4.2.

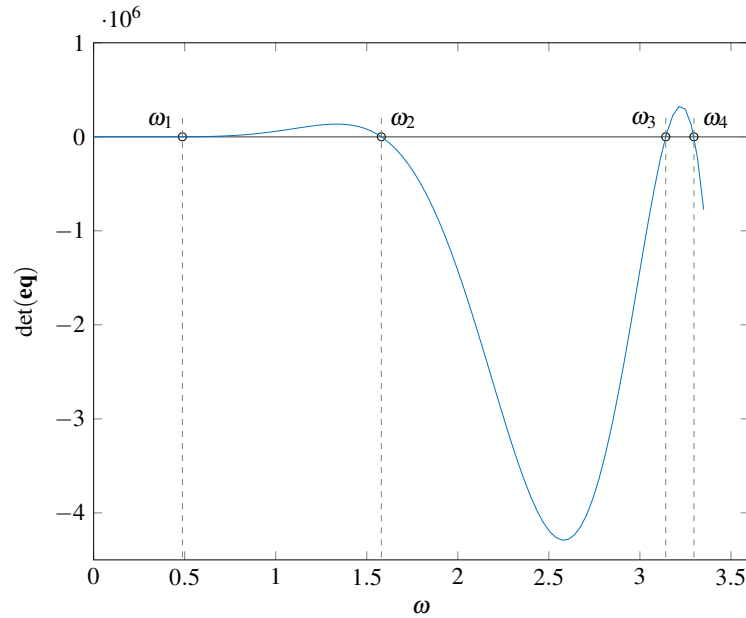


Figure 4.5.: Characteristic function for a single clamped-pinned beam.

Table 4.2.: The natural frequencies of the clamped-pinned beam in [rad/s].

Nr. of eigenfrequency	Euler-Bernoulli	Timoshenko	FEM
1 (transv.)	0.487566	0.477197	0.4848
2 (transv.)	1.580028	1.489210	1.5476
3 (long.)	3.141593	3.141593	3.1552
4 (transv.)	3.296602	2.953855	3.1576

The first four eigenmodes can be studied in Figure 4.6 and Figure 4.7. Furthermore, the qualitative pass of the internal forces can be observed in Figure 4.8.

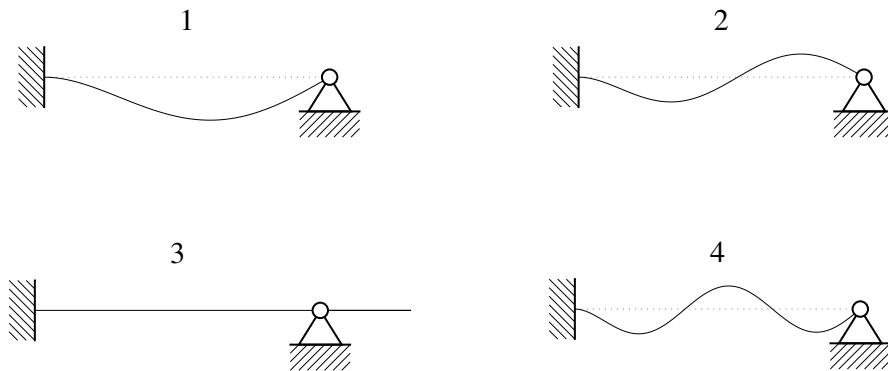
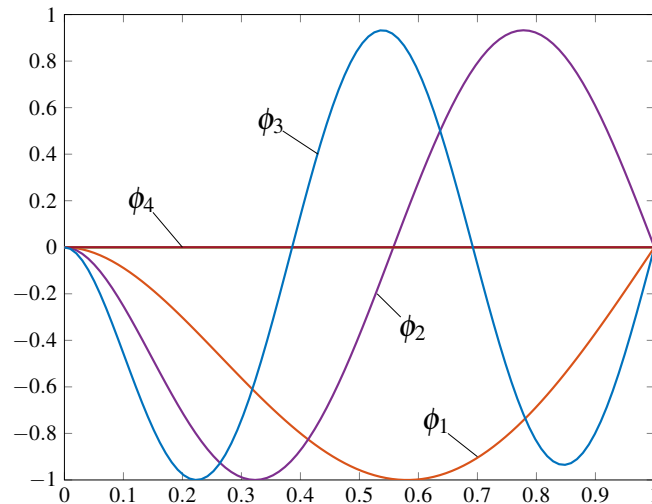


Figure 4.6.: The first four eigenmodes of a clamped-pinned beam.

Figure 4.7.: The eigenmodes ϕ_i , $i = \{1, 2, \dots, 4\}$ of a clamped-pinned beam in one plot.

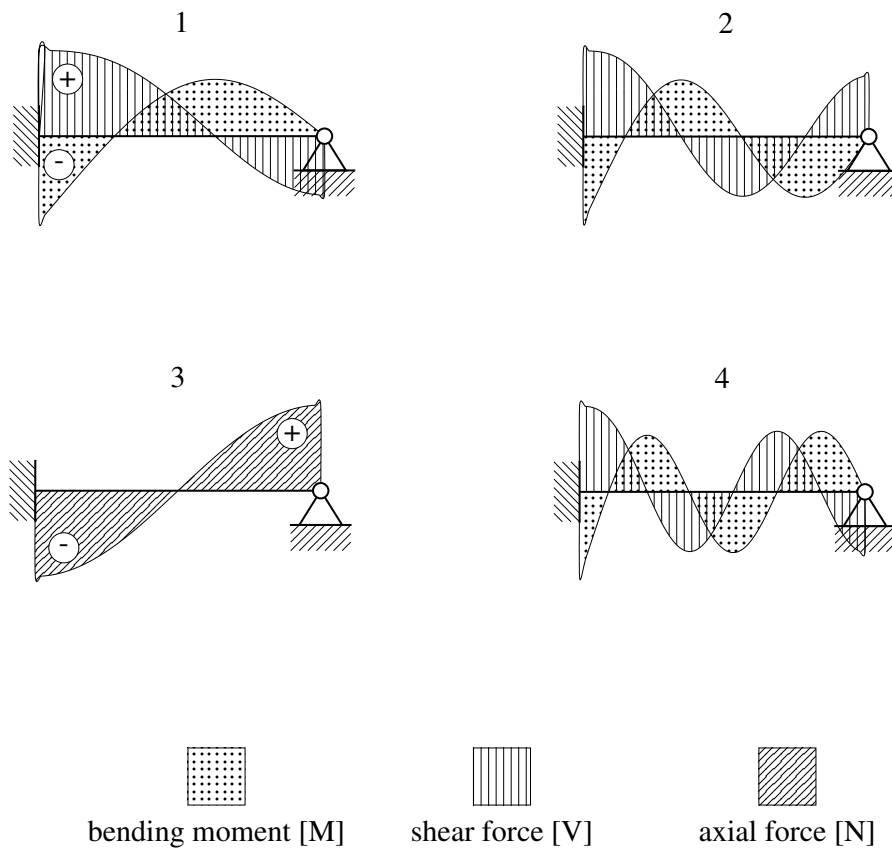
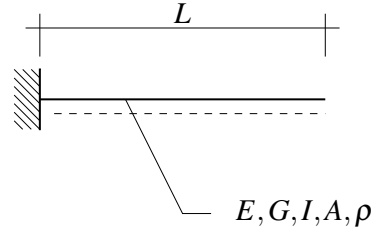


Figure 4.8.: The quantitative internal forces of a clamped-pinned beam occurring at the natural frequencies 1-4.

4.3. Cantilever Beam

The cantilever beam example was executed under the following general parameters

$$\begin{aligned}
 E &= 1 && [kN/m^2] \\
 G &= 1 && [kN/m^2] \\
 I &= 1 && [m^4] \\
 A &= 1 \cdot 10^3 && [m^2] \\
 L &= 1 && [m] \\
 \rho &= 1 && [t/m^3] .
 \end{aligned}$$



Hand calculation

The cantilever beam is a common structure for engineers, therefore the availability of analytic formulas to calculate the transversal eigenfrequencies by hand is given for the Euler-Bernoulli beam theory [4]. To get the first three transversal natural frequencies, the equation

$$\omega_i = (\lambda_i L)^2 \sqrt{\frac{B}{\mu L^4}} \quad (4.10)$$

was used, which can be specified for this case as

$$\omega_i = \sqrt{\frac{\lambda_i^4}{1 \cdot 10^3}} \quad (4.11)$$

λ_i defines the i^{th} eigenvalue and $B = EI$ the bending stiffness. For common structures as the cantilever beam, the eigenvalues are often pre-calculated and available in the literature.

Table 4.3.: λ_i and ω_i for a cantilever beam.

Nr. of transv. eigenmode	Eigenvalue λ_i	Natural frequency ω_i
1	1.875	0.111174
2	4.694	0.696764
3	7.855	1.951158

To get a comparison, the cantilever beam was modelled with the finite element code using ten elements. The results are added in the column 'FEM' in Table 4.4.

For the longitudinal eigenfrequencies, the general solution

$$U_{(x)} = C_1 \cos\left(\frac{\omega}{c}x\right) + C_2 \sin\left(\frac{\omega}{c}x\right), \quad (4.12)$$

can be used. Enforcing the boundary conditions

$$\begin{aligned} U_{(x=0)} = 0 & \rightarrow C_1 = 0, \\ N_{(x=L)} = EA U'_{(x=L)} = 0 & \rightarrow C_2 \cos\left(\frac{\omega}{c}L\right) = 0 \end{aligned}$$

leads to

$$\cos\left(\frac{\omega}{c}L\right) = 0 \rightarrow \frac{\omega}{c}L = \frac{\pi}{2} + n_f\pi \rightarrow \omega_{n_f} = \frac{c}{L} \left(n_f\pi - \frac{\pi}{2}\right).$$

Considering the wave propagation velocity $c = \rho E$, the first longitudinal eigenfrequency can be calculated as

$$\omega_{l,1} = \frac{\rho E}{L} \frac{\pi}{2} = \frac{\pi}{2} = 1.570796 \text{ rad/s}. \quad (4.13)$$

Computation using the NAT

The solution procedure was applied to this structure leading to the characteristic function Equation (4.14) and its natural frequencies based on the Euler-Bernoulli beam theory in Table 4.4.

$$f(\omega) = 2AE^3 I^2 \omega^4 \rho^2 \alpha (\delta)^{3/2} \cos(\beta) e^{-2\gamma} [\cos(\gamma) + 2e^\gamma + e^{2\gamma} \cos(\gamma)] \quad (4.14)$$

where

$$\begin{aligned} \alpha &= \sqrt{\frac{\rho}{E}} & \delta &= \frac{A\rho}{EI} \\ \beta &= L\omega\alpha & \gamma &= L\sqrt[4]{\omega^2\delta}. \end{aligned}$$

To obtain the natural frequencies it holds $f(\omega) \stackrel{!}{=} 0$. A plot of the frequency function can be seen in Figure 4.9. The natural frequencies of the cantilever beam are collected in Table 4.4 and the associated mode shapes are shown in Figure 4.10 and Figure 4.11. The qualitative internal forces can be observed in Figure 4.12.

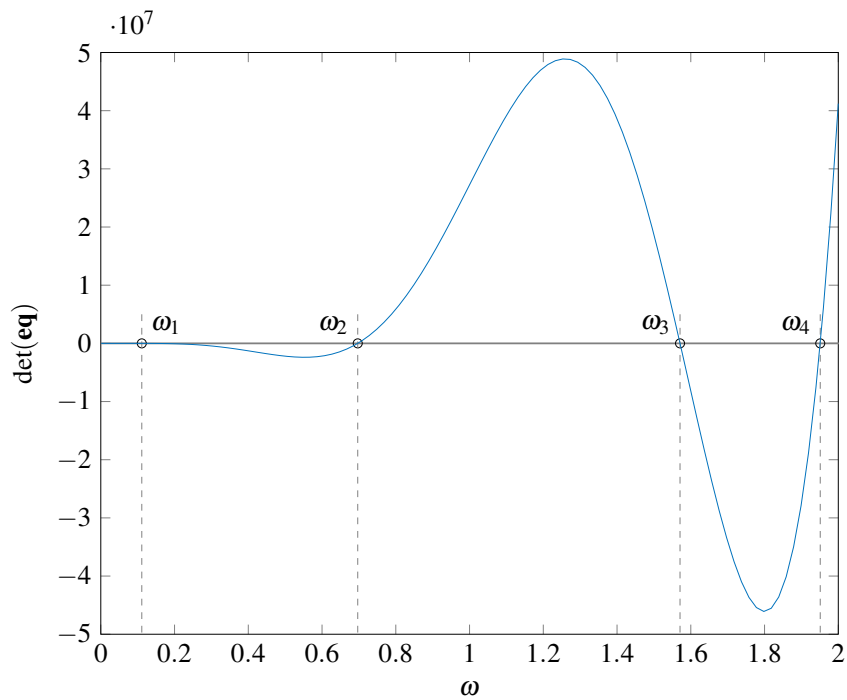


Figure 4.9.: Characteristic function for a cantilever beam.

Table 4.4.: The natural frequencies of the cantilever beam in [rad/s].

Number of eigenfrequency	Euler-Bernoulli computation	Timoshenko computation	Euler-Bernoulli hand calculation	FEM
1 (transv.)	0.111186	0.110623	0.111174	0.1109
2 (transv.)	0.696792	0.673312	0.696764	0.6858
3 (long.)	1.570796	1.570796	1.570796	1.570796
4 (transv.)	1.951037	1.806631	1.951158	1.8804.

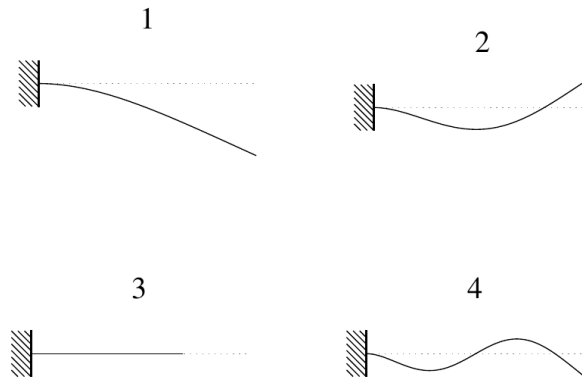


Figure 4.10.: The eigenmodes of a cantilever beam.

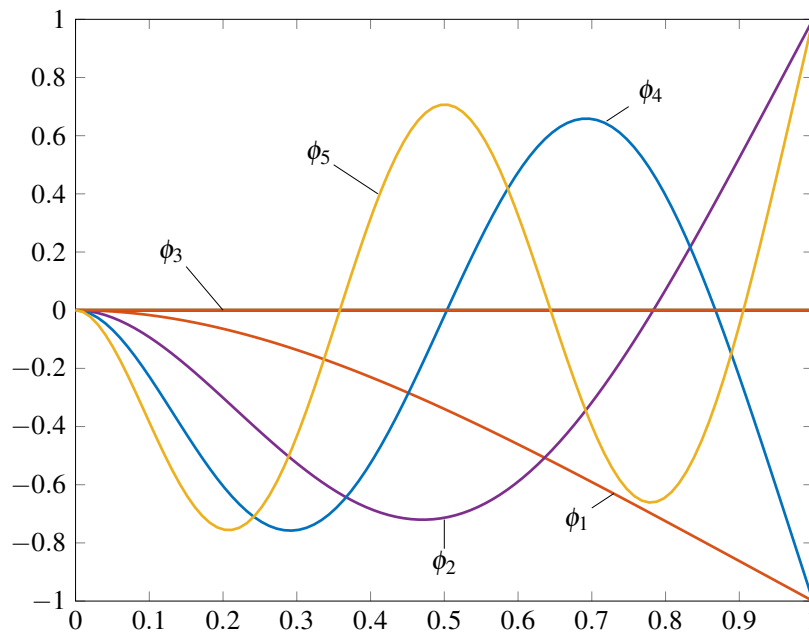


Figure 4.11.: The eigenmodes $\phi_i, i = \{1, 2, \dots, 5\}$ of a cantilever beam in one plot.

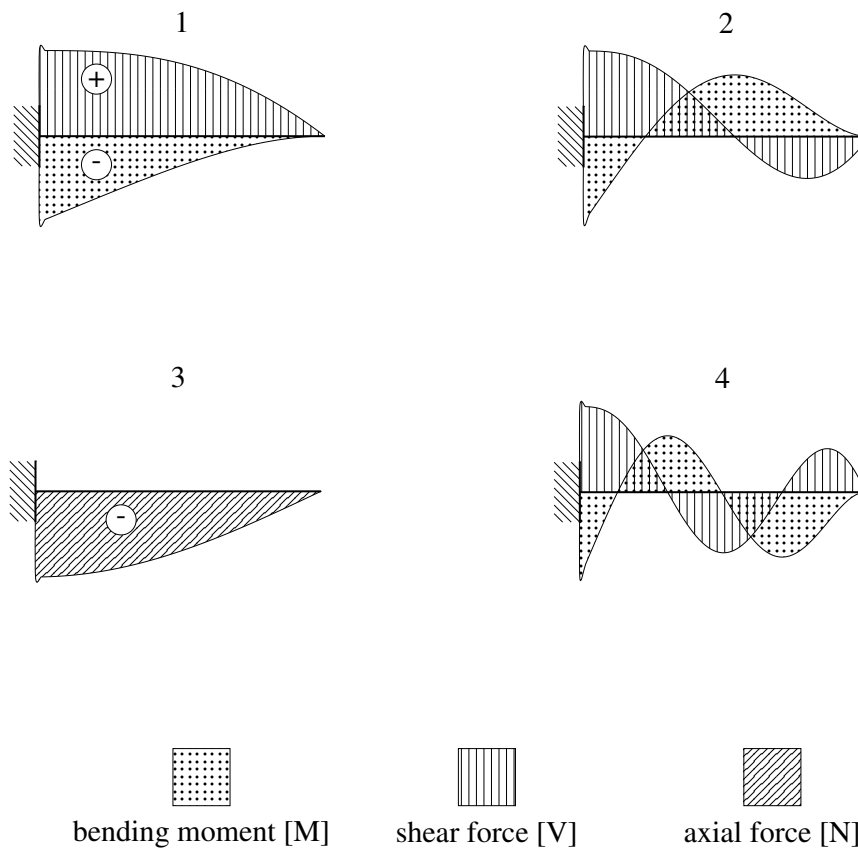
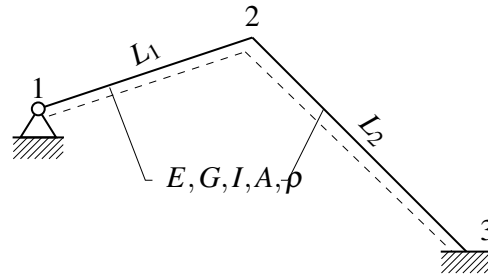


Figure 4.12.: The quantitative internal forces of a cantilever beam occurring at the natural frequencies 1-4.

4.4. Two-Beam Frame

In the following example, two beams are rigidly connected and forming an arbitrary frame structure. The frame is pinned on one side and fixed on the other side. The two-beam frame example was executed under the following general parameters for both beams

$$\begin{aligned} E &= 1 \quad [kN/m^2] \\ G &= 1 \quad [kN/m^2] \\ I &= 1 \quad [m^4] \\ A &= 1 \quad [m^2] \\ \rho &= 1 \quad [t/m^3]. \end{aligned}$$



N_τ	$[X_{N_\tau}, Y_{N_\tau}]$
1	[0,0]
2	[3,1]
3	[6,-2]

L_m	[m]
L_1	3.1623
L_2	4.2426

Each frame node N_τ is defined by the coordinates $[X_{N_\tau}, Y_{N_\tau}]$. Through the connectivity of the frame nodes N_τ frame members $m \in \tau$ are formed. The length of each frame member L_m is determined within the computation.

Finite Element Computation

No simple formulas to estimate the natural frequencies for such arbitrary frame structures are available. Therefore, the frame was modelled in the finite element-based code.

Analytical Computation using the NAT

The solution procedure was applied to this structure and the results are added in Table 4.5. In the plot of the frequency function Figure 4.13 it can be noted that the amplitudes between each natural frequency increase drastically. Thus, when plotting the characteristic function over a broad frequency range, the smaller amplitudes are no longer visible. Therefore, the frequency function was plotted on a logarithmic scale, see Figure 4.14.

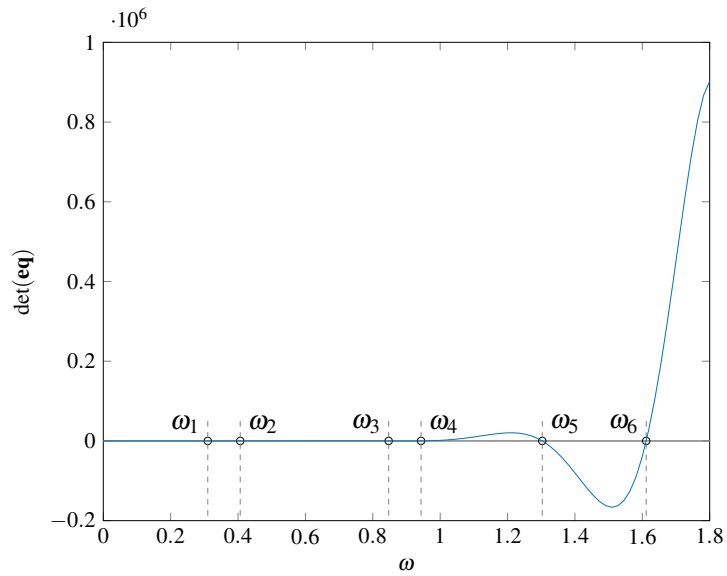


Figure 4.13.: Characteristic function $f(\omega)$ for the two-beam frame.

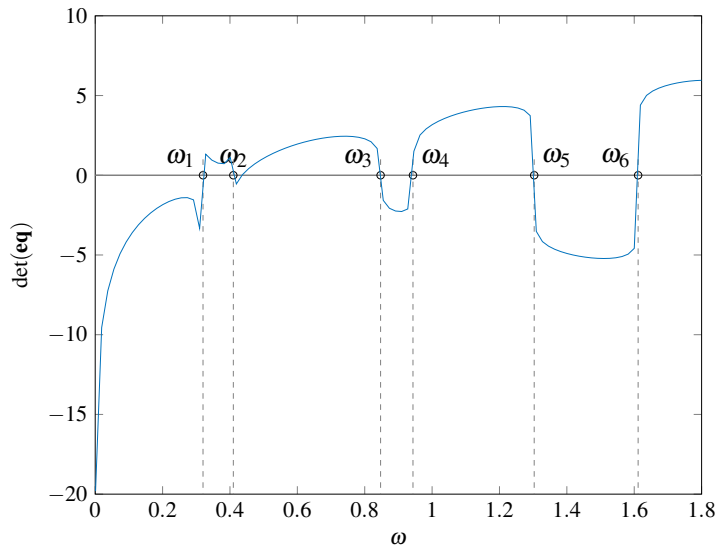
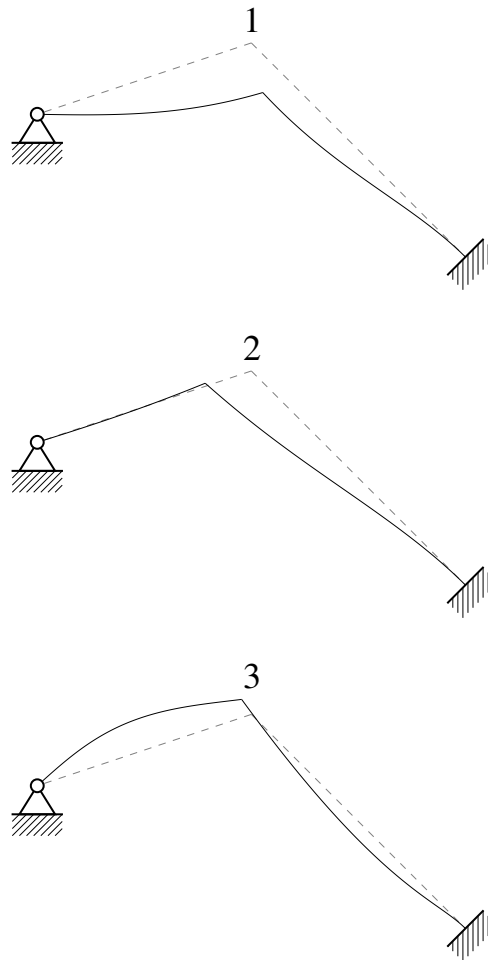


Figure 4.14.: Characteristic function $f(\omega)$ on a logarithmic scale.

Table 4.5.: The natural frequencies of the two-beam frame in [rad/s].

Nr. of eigenfrequency	Euler-Bernoulli	Timoshenko	FEM
1	0.310142	0.104480	0.316
2	0.406412	0.360978	0.427
3	0.847092	0.767214	1.212

Figure 4.15.: The mode shapes ϕ_i , $i = \{1, 2, 3\}$ of the two-beam frame.

5. PRACTICAL EXAMPLES

In comparison to the Test Examples which serve to verify the reliability of the computational code, in this chapter, the capability of the code is tested. Therefore more complex structures are modelled, using an increased number of frame members. Structures where the dynamic analysis is of importance can be e.g. a railway bridge modelled as a metallic truss system or a wooden frame embodying a multi-storey building. Computing the natural frequencies and mode shapes of those examples can show if the code's potential suits practical modelling.

5.1. Multi-Storey Frame

Tall buildings exposed to earthquakes or other dynamic forces constitute a danger to people in and around the structure. To prevent those buildings from much damage the natural frequencies need to be determined. The structural parameters are chosen for the wooden frame structure as $E_m = 12.5 \cdot 10^6 [kN/m^2]$, $\rho_m = 0.41 [t/m^3]$, $A_m = 8 \cdot 10^{-2} [m^2]$ and $I_m = 11 \cdot 10^{-4} [m^4]$ and a sketch of the structure can be regarded in Figure 5.1.

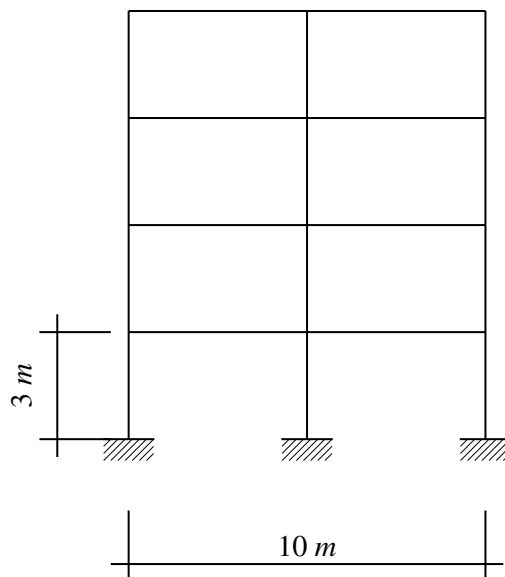


Figure 5.1.: The sketch of the multi storey frame.

The wooden multi-storey building is modelled as a frame structure. Hence, the frame members are connected rigidly and the transversal vibrations are dominant. Therefore, it is assumed that the choice of the beam theory matters in the way that the frame using the Euler-Bernoulli beam theory appears to be stiffer than processing with the Timoshenko beam theory. The first lateral natural frequency occurs at $37.808498 \text{ [rad/s]} = 6.0174 \text{ [Hz]}$ regarding the Euler-Bernoulli- and at $32.4654 \text{ [rad/s]} = 5.1670 \text{ [Hz]}$ respecting the Timoshenko beam theory and provokes the first horizontal mode shape shown in Figure 5.2. It is comprehensible that the system acts 'softer' when considering the Timoshenko beam theory due to the additional shear deformation. The comparison of the results between the two different beam theories and the finite element based computation can be studied in Table 5.1. Besides the superior behaviour of wooden structures due to earthquakes, the footfall sound can be seen as a weak point. Also acoustic problems need a dynamic analysis, whereas the focus there is on a higher frequency range.

Table 5.1.: The first natural frequency of the multi-storey frame in [rad/s].

Euler-Bernoulli	Timoshenko	FEM
37.808498	32.4654	37.452

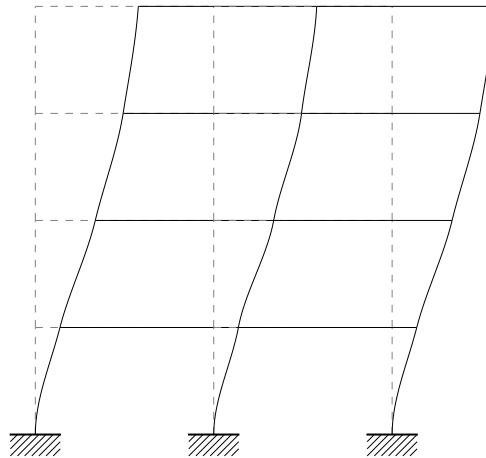


Figure 5.2.: The first lateral eigenmode of the multi storey frame.

5.2. Railway Bridge

Truss systems can also be modelled and their natural frequencies and mode shapes can be computed. Dynamic observations of bridges are essential, not only due to the dynamic excitation through moving vehicles but also to determine the long-term fatigue of the structure. The parameters for this example are $E_m = 2 \cdot 10^8 \text{ [kN/m}^2\text{]}$, $G_m = 0.8 \cdot 10^{14}$, $A_m = 0.01 \text{ [m}^2\text{]}$, $I_m = 3.5 \cdot 10^{-4} \text{ m}^4$ and $\rho_m = 7.85 \text{ [t/m}^3\text{]}$ for each truss member. A visualization of the structure can be seen in Figure 5.3 (top).

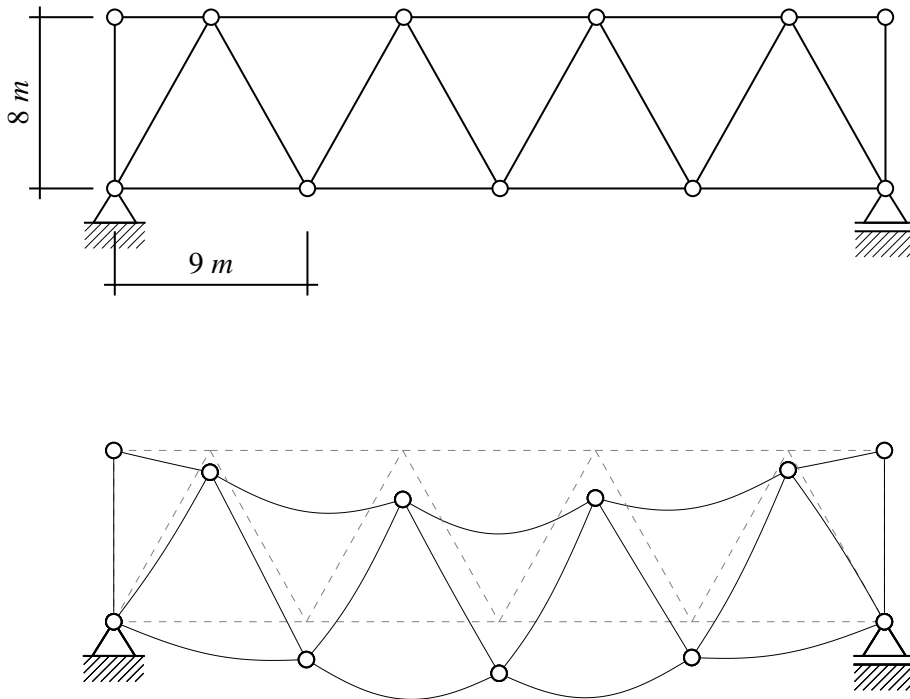


Figure 5.3.: The sketch (top) and the first mode shape (bottom) of the railway bridge.

Table 5.2.: The first natural frequency of the railway bridge in [rad/s].

Euler-Bernoulli	Timoshenko	FEM
69.49832	69.4523	70.269

Due to the fact that bending moments can not be transmitted over hinges, longitudinal vibrations dominate when observing a truss system. Hence, no significant differences of the natural frequencies are expected when changing between the two beam theories. Accordingly, the first natural frequency of this bridge was computed as $\omega = 69.498320 \text{ [rad/s]} =$

11.061001 [Hz] for the Euler-Bernoulli beam theory and $\omega = 69.4523$ [rad/s] = 11.0537 [Hz] respecting the Timoshenko beam theory. The natural frequencies are gathered in Table 5.2 and the corresponding mode shape is shown in Figure 5.3 (bottom).

To sum up, it can be noted that the NAT code is capable of processing more complex structures. Although the computational time rises as more frame members are considered, the calculation process is stable, and the results are accurate. Building the determinant of the system matrix and finding the zeros of that determinant turn out to be the main two steps causing the increase in computational time.

6. IMPLEMENTATION

The MATLAB code was created to enable dynamic analysis of 2D-frames with any shape and supports. Several frame members can be connected with a range of hinges and several bearings can be applied on the frame nodes N_τ . This project aims to obtain the eigenfrequencies of the chosen frame structure analytically based on two different beam theories such as the Euler-Bernoulli- and Timoshenko- beam theory. Also, the corresponding mode shapes are determined. The code was created in MATLAB and separated into the main function and several sub-files and functions. The hierarchy of code files can be seen in Figure 6.1.

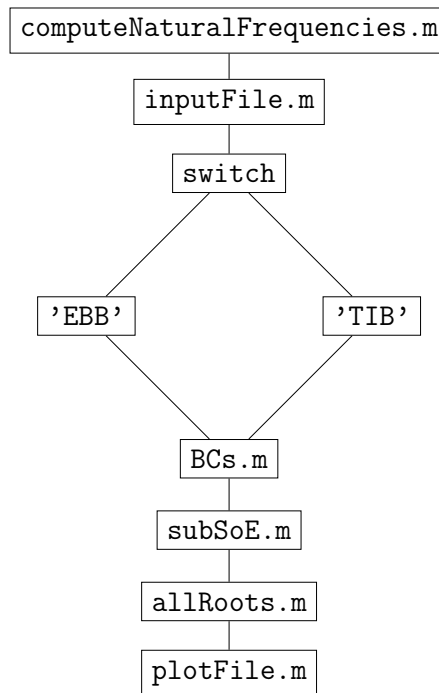


Figure 6.1.: The hierarchy of code files.

To get a better insight into the computational process, a more detailed flow chart was made, see Figure 6.2. This should help to visualize the computational steps. Furthermore, each part of the code is described separately.

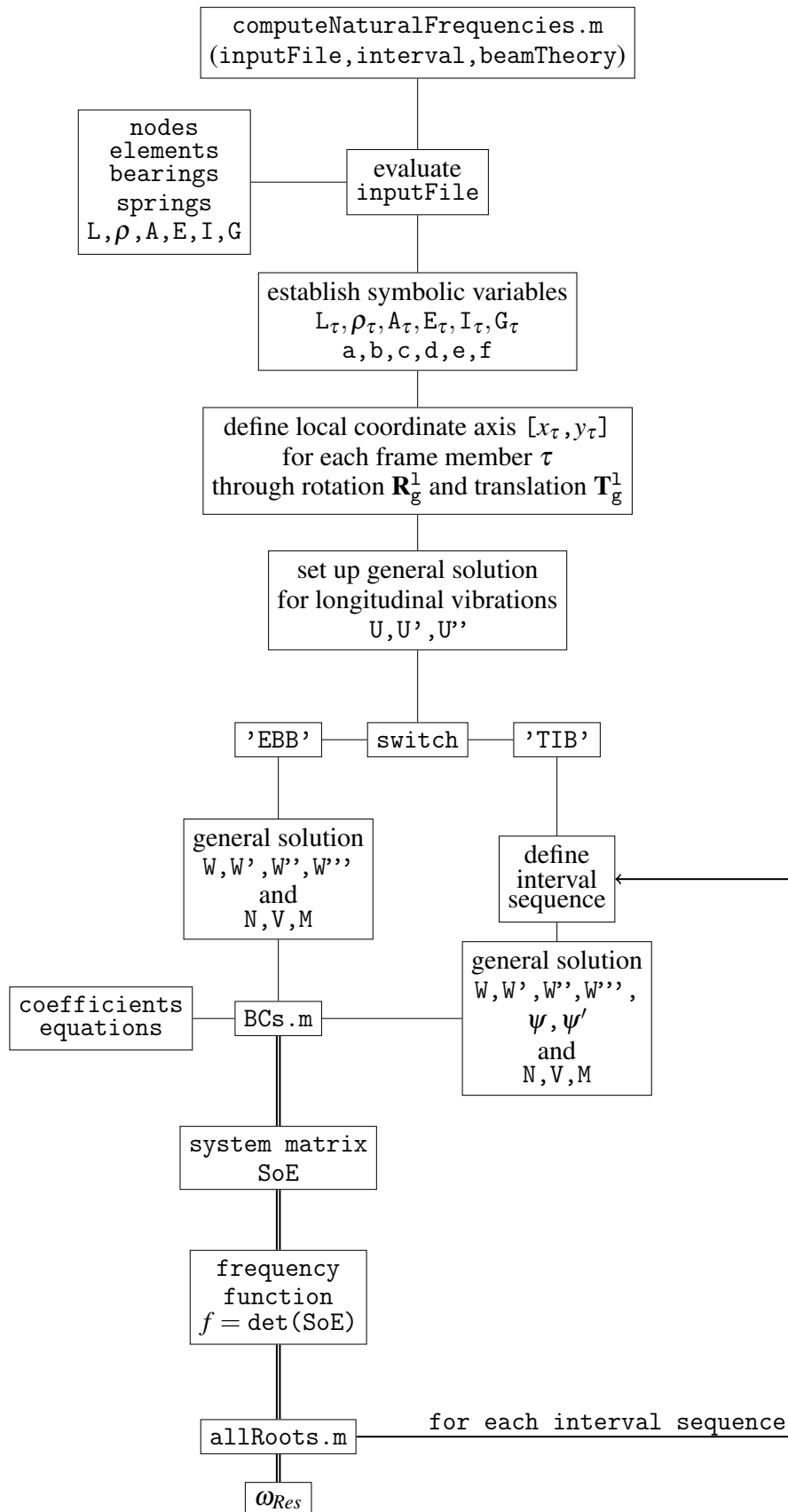


Figure 6.2.: The computation process to calculate the natural frequencies.

Nomenclature regarding the code		
i	subscript for $i = \{1 : nBeams\}$	
L	symbolic unit length 1m	
$nBeams$	number of frame members	
<u>symbolic variables</u>	<u>actual parameters</u>	
x_i	the axial coordinate	X_i
l_i	length of the beam	$lBeam_i$
A_i	cross-section area	$CrossSectionArea_i$
E_i	Young's Modulus	$YoungsModulus_i$
I_i	moment of inertia	$MomentOfInertia_i$
G_i	shear modulus	$ShearModulus_i$
ρ_i	material density	$MaterialDensity$
ω	circular frequency	$naturalFrequencies$
c	spring stiffness	$springConstant$
$a_i - f_i$	integration constants	
x_i, y_i	local system coordinates	
u_i	longitudinal deformation in frequency domain	
w_i	bending deflection in frequency domain	
ϕ_i	rotational degree of freedom	
V_i	shear force	
M_i	rotational moment	
N_i	axial force	
U_i	total deflection in $[X, Y]$	
S_i	internal forces in $[X, Y]$	
ev	matrix of eigenvectors	
ew	matrix of eigenvalues	
EBB	Euler-Bernoulli beam theory	
TIB	Timoshenko beam theory	

6.1. computeNaturalFrequencies.m

The file `computeNaturalFrequencies.m` accounts as the main function to evaluate the natural frequencies along with the corresponding mode shapes. Within this function multiple other files and functions are evaluated, see Figure 6.2. The main function

`computeNaturalFrequencies.m` can be ran through a MATLAB command, e.g.

```
omega = computeNaturalFrequencies("inputFileName",[interval],"beamTheory").
```

The input parameters are essential for the computations, further details will be given in Sections 6.2 to 6.5. Anyway, a possible command to start the computation of the natural frequencies and mode shapes in a range of 0 – 10 [rad/s] for a Euler-Bernoulli frame structure with multiple frame members, would be

```
omega = computeNaturalFrequencies("multiBeamFrame",[0 10],"EBB").
```

The explanation of the code will be executed based on an example of an arbitrary frame structure, see the "multiBeamFrame" in Figure 6.3. How to obtain the natural frequencies of this arbitrary structure will be explained step by step by showing parts of the code.

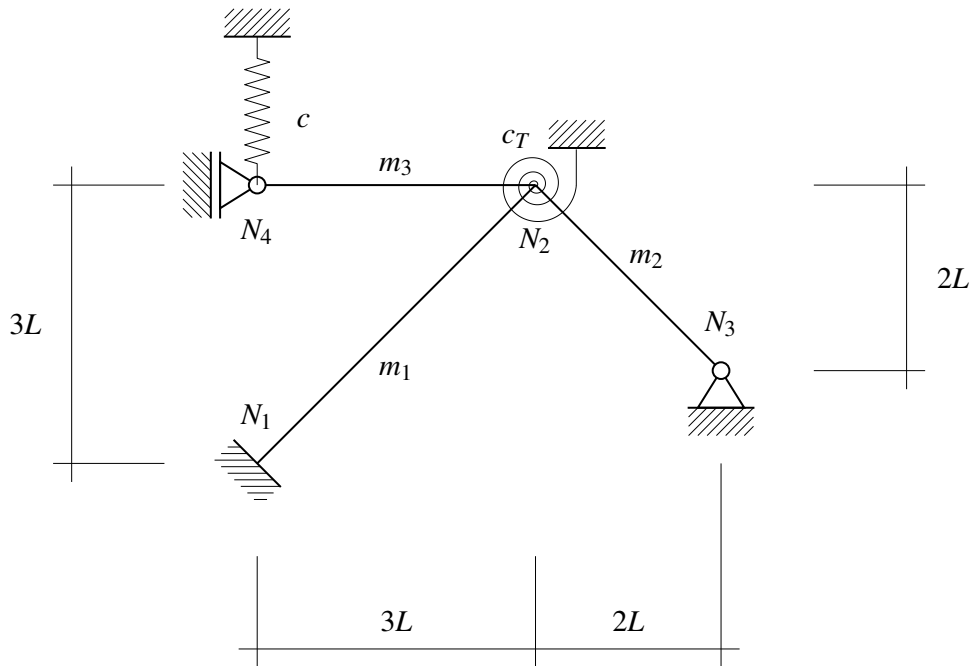


Figure 6.3.: The system plot of the multiple beam frame.

The first lines define the input data of the function, see Listing 6.1, line 1-2. Furthermore, the file paths of potential input files are added. Thus, the `inputFile.m` is evaluated, see Listing 6.1, line 6. Meaning, that all defined parameters of the structure, as well as the matrices containing the nodes and elements established in the `inputFile.m` are added as variables to the workspace.

```

1 function naturalFrequencies = computeNaturalFrequencies(
    inputFileName, interval, theory)
2
3 addpath(genpath(pwd))
4
5 %% define variables
6 eval(inputFileName)

```

Listing 6.1.: Call of the function and evaluation of the input file.

Define variables

Although, the structural variables are available, the computations are processed using symbolic variables $l_i, A_i, I_i, G_i, E_i, \rho_i$, Listing 6.2, line 7-12, as well as x and ω , see Listing 6.1, line 3 being the axial coordinate and rotational frequency respectively. This should help retracing and understanding the computational steps and enables symbolic calculations up to the point when the zeros of the frequency function need to be determined. Thus, before finding the zeros of the characteristic function $f(\omega)$, the actual parameters need to be embedded. Also variables for the integration constants a, b, c, d, e, f , are required for each frame member, see Listing 6.2, line 4-5.

```

1
2 nBeams = size(elements,1);
3 syms x omega
4 coeffs = [sym('a',[nBeams 1]), sym('b',[nBeams 1]), sym('c',[nBeams
    1])...
5           , sym('d',[nBeams 1]), sym('e',[nBeams 1]), sym('f',[nBeams
    1])];
6
7 l = sym('l',[nBeams 1]);
8 A = sym('A',[nBeams 1]);
9 E = sym('E',[nBeams 1]);
10 I = sym('I',[nBeams 1]);
11 G = sym('G',[nBeams 1]);
12 rho = sym('rho',[nBeams 1]);

```

Listing 6.2.: Setup of the symbolic variables for the computation.

Local coordinates

Each frame member will count as the domain where the established general solution for longitudinal and transversal vibrations holds. For frame members that are modelled in an angle at an arbitrary location in the two-dimensional space, its local coordinate system needs to occur a shift and a rotation, see Listing 6.3, line 11-12. Therefore, the global unit vectors $[X_{g,0,m}, Y_{g,0,m}]$ are transferred from the origin to the start node $k_{m,0}$ of the frame

member and rotated using the rotation matrix, see Listing 6.3, line 7-10 and (6.1). Anyway, when plotting, the deformation for each frame member in the local coordinates will be transferred back to the global coordinate system in order to use the plot function.

$$\mathbf{R}_g^1 = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (6.1)$$

```

1 %% rotated coordinate system
2 eAxes = [1 0;0 -1]; %ex ez
3 nodesTex = subs(nodes,L,1);
4 for i=1:nBeams
5     lBeam(i) = sqrt(sum((nodes(elements(i,2),:)-nodes(elements(i,1),:)).^2)); % Beam lengths
6     % transformation of coordinate system
7     lclOrig(i,:) = nodesTex(elements(i,1),:); % origin, local css
8     gCoord(i,:) = nodes(elements(i,2),:)-nodes(elements(i,1),:); %
9     Vectors in global coordinates
10    rotAn(i,:) = atan2(subs(gCoord(i,2),L,1),subs(gCoord(i,1),L,1));
11    % rot angle for axis
12    rotMat(:,:,i) = [cos(rotAn(i,1)), -sin(rotAn(i,1)); sin(rotAn(i,1)), cos(rotAn(i,1))]; %rotation matrix
13    lclAxes(:,:,i) = rotMat(:,:,i)*eAxes; % rotated local axes
14    LCLAxes(:,:,i) = double(lclOrig(i,:))+double(lclAxes(:,:,i)); %
15    rotated + shifted local axes
16 end

```

Listing 6.3.: Shift and rotation of the unit vectors $X_{g,0,m}, Y_{g,0,m}$

Longitudinal vibrations

The general solution for longitudinal vibrations (6.2) can be set up for each frame member since the differentiation in beam theories only applies to transversal vibrations the longitudinal Ansatz holds for both cases, see Listing 6.4.

$$u_m = e \cos(\lambda_m x) + f \sin(\lambda_m x) \quad \text{with} \quad \lambda_m = \omega \sqrt{\frac{\rho_m}{E_m}} \quad (6.2)$$

```

1 %% longitudinal vibrations
2 lambda = omega .* sqrt(rho./E);
3 u = coeffs(:,5) .* cos(lambda.*x) + coeffs(:,6) .* sin(lambda.*x);
4 du = diff(u,x);
5 ddu = diff(u,x,x);

```

Listing 6.4.: Introducing the general solution for longitudinal vibrations.

Transversal vibrations / Euler-Bernoulli

Depending on the chosen beam theory, the code switches at this point. After the computation of the natural frequencies, the code reunites again to further plot the results. Here, the general solution for the transversal deformation regarding the Euler-Bernoulli beam (6.3) is implemented and its derivatives with respect to the local beam axis x are formulated, see Listing 6.5, line 3-9.

$$W_m = a \cos(\kappa_m x) + b \sin(\kappa_m x) + c \cosh(\kappa_m x) + d \sinh(\kappa_m x) \quad (6.3)$$

$$\text{with } \kappa_m = \sqrt[4]{\frac{\omega^2 \rho A_m}{E_m I_m}}.$$

```

1 case 'EBB'
2 %% definition Bernoulli
3 % transversal vibrations
4 kappa = sqrt(omega) .* ((rho.*A)./(E.*I)).^(1/4);
5
6 w = coeffs(:,1) .* cos(kappa.*x) + coeffs(:,2) .* sin(kappa.*x)...
7     + coeffs(:,3) .* exp(kappa.*(x-1)) + coeffs(:,4) .* exp(-kappa.*
8     x);
9 dw = diff(w,x);
10 ddw = diff(w,x,x);
11 dddw = diff(w,x,x,x);
12
13 for i=1:nBeams
14 N(i,:) = E(i)* A(i)*du(i);
15 V(i,:) = -E(i)*I(i)*dddw(i);
16 M(i,:) = -E(i)*I(i)*ddw(i);
17 U(i,:) = u(i)*lclAxes(:,1,i) + w(i)*lclAxes(:,2,i); % Resultierende
18     Verformung
19 S(i,:) = N(i)*lclAxes(:,1,i) + V(i)*lclAxes(:,2,i); % Resultierende
20     Schnittkraft
21 end

```

Listing 6.5.: General solution for w kinetic and kinematic quantities

Along with the partial derivatives of w_i the kinetic quantities are defined here for the Euler-Bernoulli beam theory, where the resulting internal force S_i is the combination of the normal- and the shear force N_i and V_i respectively. Same holds for the resulting deformation U_i as it is the sum of the transversal and longitudinal deflection w_i and u_i respectively, see Listing 6.5, line 11-17.

Transversal vibrations / Timoshenko

For the Timoshenko beam, the general solution of w_i depends on the investigating frequency interval, see Listing 6.6. As the eigenvalues λ_i change, depending on the interval, from the real to the imaginary domain and vice versa. This occurs at the critical frequency ω_i^c . Since each beam has a critical frequency ω^c , the computations need to be redone for each beam and frequency interval resulting in natural frequencies within this interval.

```

1     case 'TIB'
2     %% definition Timoshenko
3     Kappa = 5/6;
4     As = A*Kappa;
5
6     for i=1:nBeams
7     frequencySwitchID(i) = sqrt(ShearModulus(i)*Kappa*CrossSectionArea(i)
8     )/MaterialDensity(i)/MomentOfInertia(i));
9
10    frequencySwitch= [interval(1) frequencySwitchID interval(2)];
11    rowsToDelete = frequencySwitch < interval(1) | frequencySwitch >
12    interval(2);
13    frequencySwitch(rowsToDelete) = [];
14    frequencySwitch = sort(frequencySwitch);
15
16    nInterval = (length(frequencySwitch)-1)+(length(frequencySwitch)-2);
17    %nInterval (Ansatz01 or Ansatz02)+nInterval (Ansatz03)
18    interval = zeros(nInterval,2);
19
20    for i=1:length(frequencySwitch)-1
21    interval(2*(i-1)+(1),:)= [frequencySwitch(i)+1e-10 frequencySwitch(i)
22    +1)-1e-10];
23
24    end
25
26    for i=1:length(frequencySwitch)-2
27    interval(2*(i-1)+(2),:)= [frequencySwitch(i+1) frequencySwitch(i+1)];
28
29    end

```

Listing 6.6.: Determining the frequency intervals for the Timoshenko beam.

As long as the observed $\omega < \omega_m^c$, all eigenvalues consist of a real and an imaginary part. Hence, the general solution is

$$W_i = a \cos(\lambda_{i,1}x) + b \sin(\lambda_{i,1}x) + c e^{-\lambda_{i,1}(x-L)} + d e^{\lambda_{i,1}x}. \quad (6.4)$$

For the case that the wanted frequency ω equals the critical frequency ω^c two eigenvalues are completely in the real domain and two are fully in the imaginary space. Thus, the general solution for this case is

$$W_i = a e^{\lambda_{i,1}x} + b e^{\lambda_{i,2}x} + c e^{-\lambda_{i,1}x} + d e^{-\lambda_{i,2}x}. \quad (6.5)$$

Upon this interval all eigenvalues $\lambda_{i,j}$ remain in the imaginary domain. Whereas, two eigenvalues consist of a real and imaginary part and the other two remain completely in the imaginary domain. As for one frame member, the general solution does not change anymore with rising frequencies. Therefore, the general solution is

$$W_i = a \cos(\lambda_{i,1}x) + b \sin(\lambda_{i,1}x) + c \cos(\lambda_{i,2}x) + d \sin(\lambda_{i,2}x). \quad (6.6)$$

The eigenvalues and the implementation of the three different general solutions are shown in Listing 6.7.

```

1 lambda1=abs(-(-(omega.*((rho.*(I.*rho.*A.^2.*E.^2*omega^2 + ...
2     4.*A.*As.^2.*E.*G.^2 - 2.*I.*rho.*A.*As.*E.*G.*omega^2 +...
3     I.*rho.*As.^2.*G.^2*omega^2))./I).^1/2) + A.*E*omega^2.*rho
4     +...
5     As.*G*omega^2.*rho)./(2.*As.*E.*G)).^1/2));
6 lambda2 = abs(-(-(A.*E.*omega^2.*rho - omega.*((rho.*(I.*rho.*A.^2.*
7     E.^2*omega^2 + ...
8     4.*A.*As.^2.*E.*G.^2 - 2.*I.*rho.*A.*As.*E.*G.*omega
9     ^2 +...
10    I.*rho.*As.^2.*G.^2*omega^2))./I).^1/2) + ...
11    As.*G*omega^2.*rho)./(2.*As.*E.*G)).^1/2));
12 naturalFrequencies=[];
13 for k=1:nInterval
14 for j=1:nBeams
15     Ansatz01=interval(k,2)<frequencySwitchID(j);
16     Ansatz02=interval(k,2)>frequencySwitchID(j);
17     Ansatz03=interval(k,2)==frequencySwitchID(j);
18 if Ansatz01
19 w(j,:) = coeffs(j,1) * cos(lambda1(j)*x) + coeffs(j,2) * sin(lambda1
20 (j)*x)...
21 + coeffs(j,3) * exp(-lambda1(j)*(x-l(j))) + coeffs(j,4) * exp(
22 lambda1(j)*x);
23 elseif Ansatz02
24 w(j,:) = coeffs(j,1) * cos(lambda1(j)*x) + coeffs(j,2) * sin(lambda1
25 (j)*x)+...
26 coeffs(j,3) * cos(lambda2(j)*x) + coeffs(j,4) * sin(lambda2(j)*x
27 );
28 elseif Ansatz03
29 w(j,:) = coeffs(j,1) * exp(lambda1(j)*x) + coeffs(j,2) * exp(lambda2
30 (j)*x)...
31 + coeffs(j,3) * exp(-lambda1(j)*x) + coeffs(j,4) * exp(-lambda2(
32 j)*x);

```

26 `end`
27 `end`

Listing 6.7.: The eigenvalues and the three different cases of general solutions.

Depended on the general solution for w the general solution for the rotational degree of freedom is

$$\psi_i = -\frac{w_i''' + w_i \left(\frac{\omega^2 \rho_i}{G_i K_i} + \frac{A_{s,i} G_i}{E_i I_i} \right)}{\frac{\omega^2 \rho_i}{E_i} \frac{A_{s,i} G_i}{E_i I_i}}. \quad (6.7)$$

For all frequency intervals, the same relation for ψ_i holds. Although, varying between different Ansatz functions for w_i will change ψ_i as well as M_i, V_i, U_i and S_i depending on the frequency interval, see Listing 6.8.

```

1 dw = diff(w,x);
2 ddw = diff(w,x,x);
3 dddw = diff(w,x,x,x);
4
5 phi = -(1./((rho.*omega^2./E)-(G*Kappa.*A./E./I))).*(dddw+(dw.*((
    omega^2.*rho./Kappa./G)+(G*Kappa.*A./E./I))));
6 dphi = diff(phi,x);
7
8 for i=1:nBeams
9 N(i,:) = E(i)* A(i)*du(i);
10 M(i,:) = E(i)*I(i)*dphi(i);
11 V(i,:) = G(i)*A(i)*Kappa*(dw(i)-phi(i));
12 U(i,:) = u(i)*lclAxes(:,1,i) + w(i)*lclAxes(:,2,i); %resulting
    deformation
13 S(i,:) = N(i)*lclAxes(:,1,i) + V(i)*lclAxes(:,2,i); %resulting
    internal force

```

Listing 6.8.: Rotational degree of freedom and the kinetic and kinematic relations.

Having the two equations W_i and ψ_i , the kinetic relations M_i, V_i can be defined, which differ in this case to the ones from the Euler-Bernoulli beam theory. All Ansatz equations are now set up and can be used to enforce the boundary conditions in the BCs.m file, see Listing 6.9, line 2, in Section 6.3 which will provide the vector of system equations eqs and the coefficient vector containing the integration constants and bearing variables cof.

Compute natural frequencies

The vector of system equations {eqs}, obtained from BCs.m, is written now in matrix form by taking out the integration constants and the variables representing the bearing constrains in {cof}, see Listing 6.9, line 5. Hence, the system equation can be written as

$$[\text{eqs}] \cdot \{\text{cof}\} = \{0\}. \quad (6.8)$$

The now called coefficient- or system matrix $[\text{eqs}]$ is simply depending on ω . For the natural frequencies of the system, it holds that the determinant of the coefficient- or system matrix needs to be zero. This is done within the `allRoots_regula_falsi` function in Listing 6.9, line 10-11. Each ω fulfilling this condition can be counted as a natural frequency.

Up to here, the computations were executed using symbolic variables. Before finding the zeros of the characteristic function, all symbolic variables need to be substituted by the structural parameters obtained from the `inputFile.m`. This substitution is done in the `subSoE.m` file, see Listing 6.9, line 7, and the code is explained in Section 6.4. Anyway, the zeros of the characteristic function, and therefore the natural frequencies are determined by the `allRoots_regula_falsi` function which is based on the Regula-Falsi method, see Listing 6.9, line 11.

```

1 %% get boundary conditions
2 BCs
3
4 %% get eigenfrequencies
5 [SoEref,~] = equationsToMatrix(eqs,cof);
6
7 subSoE
8
9 %% get eigenfrequencies
10 matlabFunction(SoEref,'File','sysMat.m');
11 naturalFrequencies = allRoots_regula_falsi(interval);

```

Listing 6.9.: Set up of system matrix and determination of natural frequencies.

Eigenvectors

The eigenvectors ϕ_n play an important role when plotting the mode shapes. It holds to find n eigenvectors that correspond to the n - computed natural frequencies. Therefore the set of eigenvalues and eigenvectors of the n^{th} system matrix $\text{SoE}_{(\omega_n)}$ needs be computed. Hence, the i^{th} column in $[\text{ew}]$ presenting the lowest eigenvalue ($\lambda \ll$) on the main diagonal, is considered as the i^{th} column in $[\text{ev}]$, responsible for the eigenvector corresponding to ω_n , see Listing 6.10.

```

1 %% get eigenvectors
2 for e = 1:nOmegaRes
3     [ev(:,:,e),ew(:,:,e)] = eig(vpa(subs(SoEref,omega,
        naturalFrequencies(e)))); %eigenvectors eigenvalues

```

```

4     [minEw(e), columnID(e)] = min(abs(diag(ew(:, :, e))));
5     evec(:, e) = ev(:, columnID(e), e);
6 end

```

Listing 6.10.: Determination of the eigenvector associating ω_n .

The natural frequencies, as well as the corresponding eigenvectors, are thus determined. Visualizing the results is done by calling the `plotFile.m` file, and will be interpreted in Section 6.5.

6.2. `inputFile.m`

The `inputFile.m` can also be renamed in case a library of structures is wished to establish. Meaning, the `inputFile.m` could be changed to e.g. `multiBeamFrame.m` as long as the `inputName` is accordingly called that way when running the function `computeNaturalFrequencies`.

In the `inputFile.m`, material and geometrical parameters E, G, ρ, A, I are defined respectively, see Listing 6.11, line 3-9. It is of importance that defining the geometry of the structure in a parametrized manner is possible. Therefore, a symbolic variable L was introduced which should be seen as a unit length of one meter, see Listing 6.11, line 1. Frame nodes are defined using multiples of L as a base for the coordinates in the two-dimensional space $[X_{N_\tau}, Y_{N_\tau}]$, see Listing 6.11, line 13. Frame members can be formed by defining the start- and end node $[k_{m,0}, k_{m,L}]$ of each frame member respectively, see Listing 6.11, line 15].

When finishing the input process, all created vectors and matrices besides `[nodes]` should have n_m rows according to the number of frame members. `[nodes]` is a $n_j \times 2$ matrix where n_j is the number of frame nodes.

```

1 syms L % symbolic variable for unit length 1m
2 %% Material
3 YoungsModulus = [1;2;3]; % young's modulus E [kN/m^2]
4 ShearModulus = [3;2;1]; % shear modulus G [kN/m^2]
5 MaterialDensity = [2;3;1]; % density rho [t/m^3]
6
7 %% Cross Section
8 CrossSectionArea = [1;10;100]; % cross sectional area A [m^2]
9 MomentOfInertia = [1;10;100]; % moment of inertia I [m^4]
10
11 %% Geometry
12 % x y
13 nodes = [-2*L, -2*L; L, L; 3*L, -L; -2*L, L];
14 % node 1 - node 2




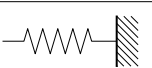
```

```
15 elements = [1,2;3,2;4,2];
```

Listing 6.11.: Definition of material and geometrical parameters frame nodes and frame members.

Bearings and springs can be attached on each frame node N_τ , see Listing 6.12, line3-6. Those frame nodes turn into supported frame nodes N_S , whereas the others remain as free frame nodes N_F . To do so the type of support needs to be chosen from the list of supports, see Listing 6.12, line 12-13. Furthermore, the frame node N_τ where the support should be attached is defined. Each support can also be implemented in an angle Φ [°] counted counter-clockwise from the initial configuration, see Table 6.1. For joints and the other supports, the rotational angle is of minor importance since it does not affect the results.

Table 6.1.: The initial configuration of bearings and springs.

roller	parallel	sliding	tr. spring
			

The scaling option is there for visualisation reasons and has no impact on the computation. When attaching springs on a frame node, see Listing 6.12, line 9-10, the spring stiffness must be defined in [kN/m] for translational springs and in [kNm/rad] for rotational springs. It should be mentioned that for one frame node N_τ , a combination of bearings, springs, and joints can be attached.

```
1 %% Bearings and Springs
2 % type - node - rotation angle [deg] (default = horizontal) - scale
3 bearings = [3,1,-45,1;
4             2,3,0,1;
5             1,4,-90,1
6             ];
7
8 % type - node - rotation angle [deg] - scale - spring stiffness [kN/
9   m] or [kNm/rad]
10 springs = [8,4,90,1,1;
11            9,2,180,1,1];
12 type = ["\Loslager", "\Festlager", "\Einspannung", "\Vollgelenk", "\
13         Halbgelenk",...
14         "\Parallelfuehrung", "\Schiebehueelse", "\Wegfeder", "\
15         Drehfeder"];

```

Listing 6.12.: Attachment of supports joints and springs at frame nodes.

To understand the parametric vectors and matrices established within the *inputFile.m* file better, all outputs based on random values are summarized.

Thus, the material parameters defined in `inputFile.m` are

$$E = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} [\text{kN/m}^2], \quad G = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} [\text{kN/m}^2], \quad \rho = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} [\text{t/m}^3] \quad \left. \vphantom{\begin{matrix} E \\ G \\ \rho \end{matrix}} \right\} \text{n- frame members,}$$

with the parameters for the cross sections

$$A = \begin{bmatrix} 1 \\ 10 \\ 100 \end{bmatrix} [\text{m}^2], \quad I = \begin{bmatrix} 1 \\ 10 \\ 100 \end{bmatrix} [\text{m}^4] \quad \left. \vphantom{\begin{matrix} A \\ I \end{matrix}} \right\} \text{n- frame members.}$$

The frame nodes N_τ of the structure are defined in the `[nodes]` matrix referring to input values in [m]. The frame members are created by connecting the frame nodes in the `[elements]` matrix.

$$\text{nodes} = \begin{array}{cc} & \begin{array}{c} X \quad Y \\ \begin{bmatrix} -2L & -2L \\ L & L \\ 3L & -L \\ -2L & L \end{bmatrix} \end{array} \\ \begin{array}{c} N_1 \\ N_2 \\ N_3 \\ N_4 \end{array} & \begin{array}{c} [m], \\ \\ \\ \end{array} \end{array} \quad \text{elements} = \begin{array}{cc} & \begin{array}{c} k_{m,0} \quad k_{m,L} \\ \begin{bmatrix} 1 & 2 \\ 3 & 2 \\ 4 & 2 \end{bmatrix} \end{array} \\ & \begin{array}{c} m_1 \\ m_2 \\ m_3 \end{array} \end{array}$$

The bearings, hinges and springs are defined through

$$\text{bearings} = \begin{array}{cc} & \begin{array}{c} \text{type} \quad \text{node} \quad \text{angle} \quad \text{scale} \\ \begin{bmatrix} 3 & 1 & -45 & 1 \\ 2 & 3 & 0 & 1 \\ 1 & 4 & -90 & 1 \end{bmatrix} \end{array} \\ & \begin{array}{c} \text{fixed} \\ \text{pinned} \\ \text{roller} \end{array} \end{array}$$

$$\text{springs} = \begin{array}{cc} & \begin{array}{c} \text{type} \quad \text{node} \quad \text{angle} \quad \text{scale} \quad \text{stiff.} \\ \begin{bmatrix} 8 & 4 & 90 & 1 & 1 \\ 9 & 2 & 180 & 1 & 1 \end{bmatrix} \end{array} \\ & \begin{array}{c} \text{transl. spring} \\ \text{rot. spring} \end{array} \end{array}$$

6.3. BCs.m

The file `BCs.m` collects all boundary and interface equations in a vector which forms the base of the system matrix. However, this vector is a composition consisting of the equilibrium of forces and moments, the bearing constraints, spring conditions, and coupling equations.

First, bearing constraints for each supported frame node N_S are gathered as variables whereas the subscripts h and v describe the horizontal and vertical component of bearing forces respectively, see Listing 6.13. Referring to the global coordinate system, this would mean the X and Y component, respectively

```

1 bearingForces = [];
2 for i = 1 : nBearings
3 bearingTyp= bearings(i,1);
4 nodeIDofBearing = bearings(i,2);
5
6 switch bearingTyp
7     case 1 %roller
8         bearingForces = [bearingForces;sym(sprintf('Fn%d',
9 nodeIDofBearing))];
10        case 2 %pinned
11            bearingForces = [bearingForces;sym(sprintf('Fh%d',
12 nodeIDofBearing)); sym(sprintf('Fv%d', nodeIDofBearing)) ];
13            case 3 %fixed
14                bearingForces = [bearingForces;sym(sprintf('Fh%d',
15 nodeIDofBearing)); sym(sprintf('Fv%d', nodeIDofBearing)); sym(
16 sprintf('M%d', nodeIDofBearing)) ];
17        case 6 %parallel guide
18            bearingForces = [bearingForces;sym(sprintf('Fn%d',
19 nodeIDofBearing)) ; sym(sprintf('M%d', nodeIDofBearing)) ];
20    end
21 end

```

Listing 6.13.: Set up of variables representing the bearing constrains.

The equilibrium conditions of forces and moments are introduced for each frame node N_τ , see Listing 6.14, line 14-15. Precaution, the sign definition needs to be taken into account, since the beam can be either positively or negatively connected to the node, see Listing 6.14, line 6-13.

```

1 %% equilibrium of forces and moments
2 for i = 1:nNodes
3     [attachedBeams,~] = find(any(elements==i,2));
4     nAttachedBeams = length(attachedBeams);
5     for j = 1:nAttachedBeams
6         isBeamPositivlyConnected = elements(attachedBeams(j),1)==i;
7         if isBeamPositivlyConnected
8             y=0;
9             fac = 1;
10        else
11            y=1(attachedBeams(j));
12            fac = -1;
13        end
14        momentenGLG(i) = momentenGLG(i) + fac*subs(M(attachedBeams(j))
15 ,x,y);

```

```

15     krafteGLG(2*(i-1)+(1:2)) = krafteGLG(2*(i-1)+(1:2)) + fac*(
      subs(S(attachedBeams(j),:),x,y)).';
16     end
17 end

```

Listing 6.14.: Equilibrium conditions of forces and moments defined on each frame node.

The bearing constraints are enforced by setting the blocked degrees of freedom at this supported frame node N_S zero, see Listing 6.15, line 19-40. It should be noted that supports as e.g. rollers are considered having a parallel and normal component that needs to be converted into the global coordinates $[X, Y]$, see Listing 6.15, line 8. This should be taken into account since bearings and springs can be attached at any angle. Again, the sign definition should be clarified before the enforcement of bearing conditions, see Listing 6.15, line 12-17.

```

1 %% bearing constrains
2 for i = 1:nBearings
3     bearingTyp=bearings(i,1);
4     [attachedToBearingBeams,~] = find(any(elements==bearings(i,2),2));
5     nAttachedToBearingBeams = length(attachedToBearingBeams);
6     nodeIDofBearing = bearings(i,2);
7     rotAngleBearing = bearings(i,3);
8     NPCComponents = [cosd(rotAngleBearing+90);sind(rotAngleBearing+90)];
9
10    for j = 1:nAttachedToBearingBeams
11
12        isBeamPositivlyConnected = elements(attachedToBearingBeams(j),1)==
            nodeIDofBearing;
13        if isBeamPositivlyConnected
14            y=0;
15        else
16            y=1(attachedToBearingBeams(j));
17        end
18
19        switch bearingTyp
20            case 1 %roller
21                bearingConstraints= [bearingConstraints; sum((subs(U(
                attachedToBearingBeams(j),:),x,y)).'*NPCComponents)) ];
22                krafteGLG(2*(nodeIDofBearing-1)+(1:2)) = krafteGLG(2*(
                nodeIDofBearing-1)+(1:2)) + ...
23                    [sym(sprintf('Fn%d', nodeIDofBearing)); sym(sprintf('
                Fn%d', nodeIDofBearing))].*NPCComponents;
24            case 2 %pinned
25                bearingConstraints= [bearingConstraints; (subs(U(
                attachedToBearingBeams(j),:),x,y)).' ];
26                krafteGLG(2*(nodeIDofBearing-1)+(1:2)) = krafteGLG(2*(
                nodeIDofBearing-1)+(1:2)) + ...
27                    [sym(sprintf('Fh%d', nodeIDofBearing)); sym(sprintf('
                Fv%d', nodeIDofBearing))];
28            case 3 %fixed

```

```

29     bearingConstraints= [bearingConstraints; (subs(U(
attachedToBearingBeams(j),:),x,y)).'];
30         subs(dw(attachedToBearingBeams(j)),x,y)];
31     krafteGLG(2*(nodeIDofBearing-1)+(1:2)) = krafteGLG(2*(
nodeIDofBearing-1)+(1:2)) + ...
32         [sym(sprintf('Fh%d', nodeIDofBearing)); sym(sprintf('Fv
%d', nodeIDofBearing))];
33     momentenGLG(nodeIDofBearing) = momentenGLG(nodeIDofBearing) +
sym(sprintf('M%d', nodeIDofBearing));
34 case 6 %parallel guide
35     bearingConstraints= [bearingConstraints; sum((subs(U(
attachedToBearingBeams(j),:),x,y)).'*NPCComponents);
36         subs(dw(attachedToBearingBeams(j)),x,y) ]);
37     krafteGLG(2*(nodeIDofBearing-1)+(1:2)) = krafteGLG(2*(
nodeIDofBearing-1)+(1:2)) + ...
38         [sym(sprintf('Fn%d', nodeIDofBearing)); sym(
sprintf('Fn%d', nodeIDofBearing))].*NPCComponents;
39     momentenGLG(nodeIDofBearing) = momentenGLG(nodeIDofBearing) +
sym(sprintf('M%d', nodeIDofBearing));
40 end
41 end
42 end

```

Listing 6.15.: Embedding the bearing constrains at each supported frame node N_g .

For all frame nodes that connect two or more frame members, the coupling equations ensure the kinematic and kinetic connectivity of the attached beams at this node, see Listing 6.16. Also here, the sign definitions need to be respected for each attached frame member, see Listing 6.16, line 9-23. Noted that besides having a rigid connection of multiple beams, also hinges, bearings, and springs can be attached in a combined matter to this frame node which influences the boundary conditions at this node, see Listing 6.16, line 24-30.

```

1 %% coupling equations
2 for i = 1:nNodes
3     [attachedBeams,~] = find(any(elements==i,2));
4     nAttachedBeams = length(attachedBeams);
5     [rowOfBearingsAtNode,~] = find(bearings(:,2)==i);
6     bearingsAtNode = bearings(rowOfBearingsAtNode,1);
7     rotationalJointAtNode=ismember(4,bearingsAtNode);
8
9     for j = 1:(nAttachedBeams-1)
10        isBeamPositivlyConnected = elements(attachedBeams(j),1)==i;
11        if isBeamPositivlyConnected
12            y=0;
13        else
14            y=1(attachedBeams(j));
15        end
16        isBeamPositivlyConnected2 = elements(attachedBeams(j+1),1)==i
;

```

```

17   if isBeamPositivlyConnected2
18       y2=0;
19   else
20       y2=1(attachedBeams(j+1));
21   end
22   couplingEquations = [couplingEquations; (subs(U(attachedBeams
(j),:),x,y) - ...
23                               subs(U(attachedBeams(j+1),:),x,y2)).')];
24   if rotationalJointAtNode
25       couplingEquations = [couplingEquations; subs(M(attachedBeams(
j)) ,x,y)];
26   else
27       couplingEquations = [couplingEquations; subs(dw(attachedBeams
(j)),x,y) - ...
28                               subs(dw(attachedBeams(j+1)),x,y2) ];
29   end
30 end
31 end

```

Listing 6.16.: Defining the coupling equations for frame nodes with multiple beams attached.

All boundary and interface equations are collected in the vector $\{eqs\}$, see Listing 6.17, line 10-13 and will be transformed into the system matrix $[eqs]$ by separating the equations from the integration constants and variables of bearing constrains gathered in the vector $\{cof\}$, see Listing 6.17, line 6-8. Thus, the system equation has the form $[eqs] \cdot \{cof\} = \{0\}$.

```

1 variable = [coeffs(:,1);coeffs(:,2);coeffs(:,3);
2             coeffs(:,4);
3             coeffs(:,5);coeffs(:,6)
4             ];
5
6 cof = [variable;
7        bearingForces
8        ];
9
10 eqs =[momentenGLG;
11        krafteGLG;
12        bearingConstraints;
13        couplingEquations];

```

Listing 6.17.: Storing the boundary equations the integration variables and the bearing coefficients in vectors.

6.4. subSoE.m

The subSoE.m file converts the system matrix [eqs] that contains the symbolic variables into the array consisting of the actual structural input values, see Listing 6.18.

```

1 % set reference length L = 1
2 SoEref = subs(SoE,L,1);
3 for i=1:nBeams
4 SoEref = subs(SoEref,l(i),lBeam(i));
5 SoEref = subs(SoEref,E(i),YoungsModulus(i));
6 SoEref = subs(SoEref,A(i),CrossSectionArea(i));
7 SoEref = subs(SoEref,G(i),ShearModulus(i));
8 SoEref = subs(SoEref,I(i),MomentOfInertia(i));
9 SoEref = subs(SoEref,rho(i),MaterialDensity(i));
10 end

```

Listing 6.18.: Substituting the symbolic variables in [eqs] with input parameters.

6.5. plotFile.m

The aim in the plotFile.m file is to visualise the results that were computed in computeNaturalFrequencies.m. The L^AT_EX environment enables plotting the results in editable .tex files and allows an easy implementation into reports. All .tex files are established in a standalone environment, ensuring a compilation of the plot by itself.

The plotFile.m file enables the visualization of mode shapes in the first instance but gives also a closer insight into the computations by showing and plotting the frequency function as well as the system matrix. Anyway, the following plots are included:

System plot in L^AT_EX

A .tex file is created to visualize the system in L^AT_EX. Vectors for each frame member are defined. Along with the coordinates, supports, hinges, and springs from the inputFile.m file the system can be plotted, see listing 6.19.

```

1 %% plot the System in latex
2 fid = fopen('../08_TeXplots/systemPlot.tex','w');
3 fprintf(fid,'%s\n','\documentclass{standalone}');
4 fprintf(fid,'%s\n','\usepackage{tikz}');
5 fprintf(fid,'%s\n','\usepackage{rotating}');
6 fprintf(fid,'%s\n','\usetikzlibrary{snakes}');
7 fprintf(fid,'%s\n','\input{../IAM_Lager.sty}');
8 fprintf(fid,'%s\n','\begin{document}');
9 fprintf(fid,'%s\n','\begin{tikzpicture}');

```

```

10 % draw beams
11 for i=1:nBeams
12 fprintf(fid, '\draw[thick] (%0.5f,%0.5f) -- (%0.5f,%0.5f);\n', ...
13     nodesTex(elements(i,1),1),nodesTex(elements(i,1),2),nodesTex(
14     elements(i,2),1),nodesTex(elements(i,2),2));
15 fprintf(fid, '\draw[->, thick,draw=red] (%0.5f,%0.5f) -- (%0.5f,%0.5
16     f) node [above]{$x_{%.0f}$};\n', ...
17     lclOrig(i,:),LCLAxes(1,:,i),i);
18 fprintf(fid, '\draw[->, thick,draw=red] (%0.5f,%0.5f) -- (%0.5f,%0.5
19     f) node [above]{$z_{%.0f}$};\n', ...
20     lclOrig(i,:),LCLAxes(2,:,i),i);
21 end
22 % draw bearings
23 for i=1:nBearings
24 fprintf(fid, "%s {(%.5f,%.5f)}[%f][%f];\n",...
25     type(bearings(i,1)), nodesTex(bearings(i,2),1),nodesTex(bearings
26     (i,2),2), bearings(i,3),...
27     bearings(i,4));
28 end
29 % draw springs
30 for i=1:nSprings
31 fprintf(fid, "%s {(%.5f,%.5f)}[%f][%f];\n",...
32     type(springs(i,1)), nodesTex(springs(i,2),1),nodesTex(springs(i
33     ,2),2), springs(i,3),...
34     springs(i,4));
35 end
36 fprintf(fid, '%s\n', '\node[above,font=\large\bfseries] at (current
37     bounding box.north) {System Plot};');
38 fprintf(fid, '%s\n', '\end{tikzpicture}');
39 fprintf(fid, '%s\n', '\end{document}');
40 fclose(fid);

```

Listing 6.19.: The system plot in \LaTeX .

System matrix and characteristic equation

Having an insight into the system matrix can help understand the code, especially with keeping the symbolic parameters in the matrix, see Listing 6.20, line 7. The same holds for the characteristic equation known as the determinant of the system matrix, see Listing 6.20, line 19. Both are implemented in a \LaTeX document.

```

1 % visualization of the system matrix in LaTeX
2 fid = fopen('../08_TeXplots/systemOfEquations.tex','w');
3 fprintf(fid, '%s\n', '\documentclass{standalone}');
4 fprintf(fid, '%s\n', '\begin{document}');
5 fprintf(fid, '%s\n', '$');
6
7 fprintf(fid, '%s %s = 0', latex(SoE), latex(variable));

```

```

8
9 fprintf(fid, '%s\n', '$');
10 fprintf(fid, '%s\n', '\end{document}');
11 fclose(fid);
12
13 display characteristic equation in LaTeX
14 fid = fopen('../08_TeXplots/characteristicEquation.tex', 'w');
15 fprintf(fid, '%s\n', '\documentclass{standalone}');
16 fprintf(fid, '%s\n', '\begin{document}');
17 fprintf(fid, '%s\n', '$');
18
19 fprintf(fid, '%s = 0', latex(simplify(detSoE)));
20
21 fprintf(fid, '%s\n', '$');
22 fprintf(fid, '%s\n', '\end{document}');
23 fclose(fid);

```

Listing 6.20.: Visualization of the system matrix and the frequency function in \LaTeX .

Plot of the frequency function

The frequency- or characteristic function $f(\omega)$ was plotted first in MATLAB, see Listing 6.21, line 6–16 and then converted into TikZ, see Listing 6.21, line 19–29, an environment especially for plots and drawings supported by \LaTeX . To do so, the `matlab2tikz` function was used, see Listing 6.21, line 16. The frequency function regarding a Timoshenko beam is a composition of n functions based on n frequency intervals. Thus, for each interval one frequency function holds.

```

1 %% characteristic function
2 % plot the characteristic function in MATLAB
3 rowsToDelete = find(interval(:,1)==interval(:,2));
4 interval(rowsToDelete,:) = [];
5 detSoEref(rowsToDelete) = [];
6 figure(1)
7 for i=1:size(interval,1)
8 omegaPlot(i,:) = linspace(interval(i,1),interval(i,2));
9 for j=1:length(omegaPlot)
10 f(i,j) = vpa(subs(detSoEref(i),omega,omegaPlot(i,j)));
11 end
12 plot(omegaPlot(i,:),f(i,:))
13 hold on
14 end
15 title('characteristic function')
16 matlab2tikz('../08_TeXplots/myfile.tex');
17
18 % plot the characteristic function in LaTeX
19 fid = fopen('../08_TeXplots/characteristicFunctionPlot.tex', 'w');
20 fprintf(fid, '%s\n', '\documentclass{standalone}');

```

```

21 fprintf(fid, '%s\n', '\usepackage{pgfplots}');
22 fprintf(fid, '%s\n', '\pgfplotsset{compat=newest}');
23 fprintf(fid, '%s\n', '\pgfplotsset{plot coordinates/math parser=false}
    ');
24 fprintf(fid, '%s\n', '\newlength\figureheight');
25 fprintf(fid, '%s\n', '\newlength\figurewidth');
26 fprintf(fid, '%s\n', '\begin{document}');
27 fprintf(fid, '%s\n', '\input{myfile.tex}');
28 fprintf(fid, '%s\n', '\end{document}');
29 fclose(fid);

```

Listing 6.21.: Plot of the frequency function in TikZ using matlab2tikz.

Plot of mode shapes

To compute the mode shapes, the symbolic vector $\{U\}$ representing the total deflection needs to be substituted with the input parameters, see Listing 6.22, line 2–8. Having a number of natural frequencies n will enforce n mode shapes. Therefore, the deflection is given in n deformation matrices resulting from n substituted natural frequencies ω at each increment of a frame member X , see Listing 6.22, line 13–14. The previously computed n eigenvectors help to get n mode shapes, in the way, that each entry of the eigenvector is replacing the integration constants and the variables of bearing constraints in $\{cof\}$, see Listing 6.22, line 15–17. The searched deflection regarding the n mode shapes $U_{n,(x)}$ can now be determined by inserting the obtained values of the eigenvectors into the equation and plotting the results for each beam and the natural frequency at each incremental point X , see Listing 6.22, line 18–28. It should be noticed that the deformation needs to be converted into the global coordinate system before plotting. Plots are done in MATLAB, see Listing 6.22, line 2–30 and L^AT_EX, see Listing 6.22, line 32–95 also including a visualisation of the structural features such as bearings, hinges and springs.

Besides the deformations, $U_{n,(x)}$ or so-called mode shapes all the other kinetic quantities such as N , V and M can also be determined for each beam and mode shape.

```

1 % plot eigenmodes in Matlab
2 for i=1:nBeams
3 U = subs(U,E(i),YoungsModulus(i));
4 U = subs(U,A(i),CrossSectionArea(i));
5 U = subs(U,I(i),MomentOfInertia(i));
6 U = subs(U,rho(i),MaterialDensity(i));
7 U = subs(U,G(i),ShearModulus(i));
8 end
9 % scale factor for plotting eigenmodes [x z]
10 scaleFac = [1e4 1];
11 for e = 1:nOmegaRes
12     figure(e+1);
13     for i = 1:size(elements,1)

```



```

14     X(i,:) = linspace(0,lBeam(i),100);
15     Ux(i,:,e) = subs(U(i,:),omega,naturalFrequencies(e));
16     for q = 1:size(variable,1)
17         Ux(i,:,e) = subs(Ux(i,:,e),variable(q),evec(q,e));
18     end
19     for j = 1:size(X,2)
20         beamUndeformedRef(j,:,i) = subs(lclOrig(i,:) + X(i,j).*
lclAxes(:,1,i)',L,1);
21         beamDeformedRef(j,:,i,e) = subs(beamUndeformedRef(j,:,i)
+ (subs(Ux(i,:,e),x,X(i,j)).*scaleFac),L,1);
22     end
23     plot(beamUndeformedRef(:,1,i),beamUndeformedRef(:,2,i),'k')
24     hold on
25     plot(beamDeformedRef(:,1,i,e),beamDeformedRef(:,2,i,e),'r')
26     hold on
27     title(sprintf('eigenmode %d',(e)))
28     grid on
29     legend off
30 end
31 end
32 %% plot eigenmodes in latex
33 fid = fopen(' ../08_TeXplots/eigenmodesPlot.tex','w');
34 fprintf(fid,'%s\n','\documentclass{standalone}');
35 fprintf(fid,'%s\n','\usepackage{tikz}');
36 fprintf(fid,'%s\n','\usepackage{rotating}');
37 fprintf(fid,'%s\n','\usetikzlibrary{snakes}');
38 fprintf(fid,'%s\n','\input{../IAM_Lager.sty}');
39 fprintf(fid,'%s\n','\begin{document}');
40 for e = 1:nOmegaRes
41     fprintf(fid,'%s\n','\begin{tikzpicture}');
42 % draw beams
43 for i=1:nBeams
44     fprintf(fid,'\draw[dotted,draw=gray] (%0.5f,%0.5f) -- (%0.5f
,%0.5f);\n', ...
45         nodesTex(elements(i,1),1),nodesTex(elements(i,1),2),nodesTex(
elements(i,2),1),nodesTex(elements(i,2),2));
46     fprintf(fid,'\draw plot [smooth] coordinates{');
47     for j=1:size(X,2)
48     fprintf(fid,' (%0.5f,%0.5f) ', ...
49         beamDeformedRef(j,1,i,e),beamDeformedRef(j,2,i,e));
50     end
51     fprintf(fid,'};\n');
52     fprintf(fid,'\draw[->,thick,draw=red] (%0.5f,%0.5f) -- (%0.5f,%0.5
f) node [above]{$x_{%.0f}$};\n', ...
53         beamDeformedRef(1,:,i,e),((beamDeformedRef(1,:,i,e)-
beamUndeformedRef(1,:,i))+LCLAxes(1,:,i)),i);
54     fprintf(fid,'\draw[->,thick,draw=red] (%0.5f,%0.5f) -- (%0.5f,%0.5
f) node [above]{$z_{%.0f}$};\n', ...
55         beamDeformedRef(1,:,i,e),((beamDeformedRef(1,:,i,e)-
beamUndeformedRef(1,:,i))+LCLAxes(2,:,i)),i);

```

```

56
57 [rowBearingAtBeam, columnBearingAtBeam] = find(elements(i,:) ==
    bearings(:,2));
58 nBearingsAtBeam = length(rowBearingAtBeam);
59 % draw bearings
60 for k=1:nBearingsAtBeam
61     isBearingAtFirstNode = columnBearingAtBeam(k)==1;
62     if isBearingAtFirstNode
63 fprintf(fid, "%s {(0.5f,0.5f)}[%f][%f];\n", ...
64         type(bearings(rowBearingAtBeam(k),1)), beamDeformedRef(1,1,i,e),
65             beamDeformedRef(1,2,i,e), bearings(rowBearingAtBeam(k),3), ...
66             bearings(rowBearingAtBeam(k),4));
67     else
68 fprintf(fid, "%s {(0.5f,0.5f)}[%f][%f];\n", ...
69         type(bearings(rowBearingAtBeam(k),1)), beamDeformedRef(end,1,i,e),
70             beamDeformedRef(end,2,i,e), bearings(rowBearingAtBeam(k),3), ...
71             bearings(rowBearingAtBeam(k),4));
72     end
73 end
74 if isempty(springs)
75 else
76 [rowSpringAtBeam, columnSpringAtBeam] = find(elements(i,:) ==springs
   (:,2));
77 nSpringsAtBeam = length(rowSpringAtBeam);
78 % draw springs
79 for k=1:nSpringsAtBeam
80     isBearingAtFirstNode = columnSpringAtBeam(k)==1;
81     if isBearingAtFirstNode
82 fprintf(fid, "%s {(0.5f,0.5f)}[%f][%f];\n", ...
83         type(springs(rowSpringAtBeam(k),1)), beamDeformedRef(1,1,i,e),
84             beamDeformedRef(1,2,i,e), springs(rowSpringAtBeam(k),3), ...
85             springs(rowSpringAtBeam(k),4));
86     else
87 fprintf(fid, "%s {(0.5f,0.5f)}[%f][%f];\n", ...
88         type(springs(rowSpringAtBeam(k),1)), beamDeformedRef(end,1,i,e),
89             beamDeformedRef(end,2,i,e), springs(rowSpringAtBeam(k),3), ...
90             springs(rowSpringAtBeam(k),4));
91     end
92 end
93 end
94 fprintf(fid, '\\node[above,font=\\large\\bfseries] at (current
    bounding box.north) {$\\omega_{res,%0.0f} = %0.2f rad/s$};\n', e,
    naturalFrequencies(e));
95 fprintf(fid, '%s\n', '\\end{tikzpicture}');
96 end
97 fprintf(fid, '%s\n', '\\end{document}');

```

Listing 6.22.: Plot of the mode shapes in MATLAB and L^AT_EX.

A summarized plot was made, gathering all mode shapes in one graph, which tend to be a practical addition when mode shapes want to be included in technical reports, see Listing 6.23.

```

1
2 %% single plot normalized mode shapes
3 figure;
4 for e = 1:nOmegaRes
5     for i = 1:size(elements,1)
6
7         maxU = max(abs(beamDeformedRef(:,2,i,e)));
8         plot(beamUndeformedRef(:,1,i),beamUndeformedRef(:,2,i))
9         hold on
10        plot(beamDeformedRef(:,1,i,e),beamDeformedRef(:,2,i,e)./maxU
11        )
12        hold on
13        %grid on
14    end
15    legendInfo{e} = [sprintf('\phi_%.0f$',e)];
16 end
17 title('mode shapes')
18 hl = legend(legendInfo);
19 set(hl, 'Interpreter','latex')
20 matlab2tikz(' ../08_TeXplots/modeShapes.tex');
21 legend off
22
23
24 fid = fopen(' ../08_TeXplots/normModeShapes.tex','w');
25 fprintf(fid,'%s\n','\documentclass{standalone}');
26 fprintf(fid,'%s\n','\usepackage{pgfplots}');
27 fprintf(fid,'%s\n','\pgfplotsset{compat=newest}');
28 fprintf(fid,'%s\n','\pgfplotsset{plot coordinates/math parser=false}
29 ');
30 fprintf(fid,'%s\n','\newlength\figureheight');
31 fprintf(fid,'%s\n','\newlength\figurewidth');
32 fprintf(fid,'%s\n','\begin{document}');
33 fprintf(fid,'%s\n','\input{modeShapes.tex}');
34 fprintf(fid,'%s\n','\end{document}');

```

Listing 6.23.: Summarized plot of mode shapes in MATLAB and L^AT_EX.

Thus, as an result the first two natural frequencies for this example of an arbitrary multiple-beam structure can be obtained in Table 6.2 along with the according mode shapes in Figure 6.4.

Table 6.2.: The first two natural frequencies of the multi-beam frame in [rad/s].

Nr. of eigenfrequency	Euler-Bernoulli	Timoshenko	FEM
1	0.10448	0.137811	0.104
2	0.360978	0.297423	0.364

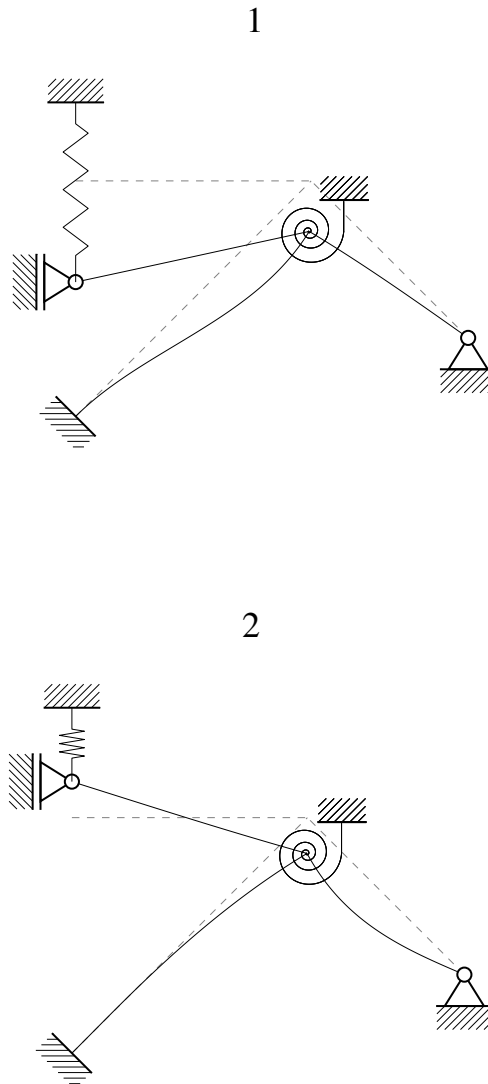


Figure 6.4.: The first two eigenmodes of the multiple beam frame.

7. CONCLUSION

Having a tool to analyse the dynamics of structures without going through the meshing process, as it is necessary for finite element programs, is practical. Another advantage besides skipping the discretization part of modelling a technical problem is the fact that analytical computations are representing the exact solution whereas finite element-based calculations refer to an approximation of the exact solution. Although the results of a finite element computation can nowadays be really accurate, there is still an open risk for miscalculation depending on the refinement of the mesh. When it comes to the fine-tuning of very sensitive technical equipment such as e.g. robotic arms etc. the exact solutions of its natural frequencies can be essential as light vibrations can already lead to failure.

However, as everything has advantages, it comes with disadvantages. One disadvantage could be the rather computational-intensive determination of the frequency function since this requires building the determinant of the system matrix that can be quite big as the number of beams forming the structure rises. A further weak point could be finding the zeros of the characteristic function. This has to be done in a numerical matter using a finite number of steps to determine the return of zero values within an interval. As the first intention was to execute all computations using symbolic variables, this goal could not be reached, since the determination of the natural frequencies requires ascertained values. Thus, every symbolic variable must have an assigned number.

Overall this method can be seen as a good alternative to existing finite element based computations and shows a different approach as to how the natural frequencies of an arbitrary structure can be determined. Not being restricted to basic beam models while still obtaining the analytical solution of a vibration problem can be seen as the main achievement of this thesis. Further upgrades of this tool could be the expansion of the computations to the three-dimensional space, including torsional vibrations, enlarging the dynamic analysis by adding plates and arcs or considering structural damping in the computation.

A. DIFFERENT APPROACH FOR THE GENERAL SOLUTION

In this thesis, hyperbolic functions are avoided in the general solution. The reason for that is the asymptotic form of this formulation as the value of x gets bigger. Whereas, the amplitudes of the exponential terms in the general solution do not exceed 1 independent from the beam length L , see Figure A.1.

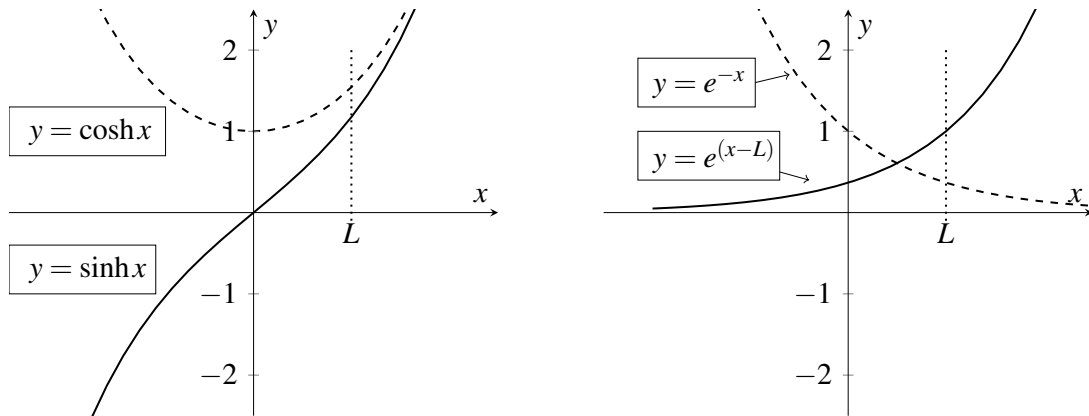


Figure A.1.: The difference of Ansatz functions for the general solution. The hyperbolic- (left) and exponential approach (right).

Due to rounding errors, small changes in the system matrix could lead to instability of the linear system when choosing the common hyperbolic functions in the general solution. The proposed exponential and trigonometrical approach ensures stable computations even at high frequencies. This can be proofed by observing the conditioning number for the system matrix $[eq]$ for the two different general solutions. The conditioning number gives information about the sensitivity of a linear system to small changes in the system matrix. The behaviour of the two different approaches can be seen in Figure A.2

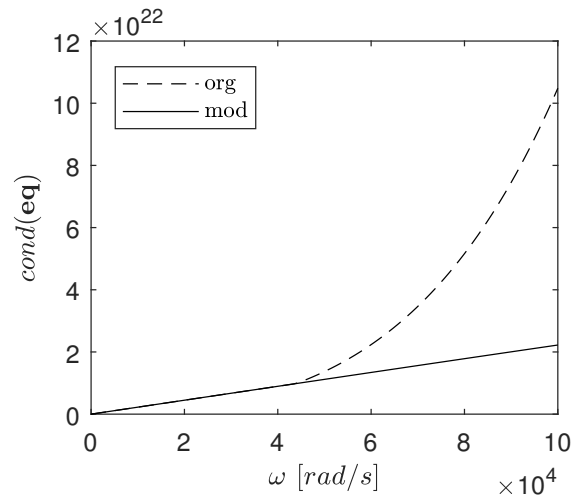


Figure A.2.: The behaviour of the conditioning number for the system matrix.

It can be observed that the linear system based on the commonly used general solution (org) as $W_{(x)} = C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 \cosh(\kappa x) + C_4 \sinh(\kappa x)$ becomes rather unstable for high frequencies. The instability of the linear system using the modified general solution (mod) as $W_{(x)} = C_1 \cos(\kappa x) + C_2 \sin(\kappa x) + C_3 e^{\kappa(x-L)} + C_4 e^{-\kappa x}$ rises linearly in a relatively moderate manner. Thus, the evaluation of natural frequencies should therefore be improved by this change and has an effect when the natural frequencies need to be accurately determined, particularly for higher frequency ranges.

B. THE BEHAVIOUR OF THE EIGENVALUES FOR A TIMOSHENKO BEAM

Starting with the equation

$$\bar{\lambda}_{1,2} = -\frac{P}{2} \pm \sqrt{\left(\frac{P}{2}\right)^2 - Q}. \quad (\text{B.1})$$

When considering the relation $\lambda = \pm\sqrt{\bar{\lambda}}$ the eigenvalue λ gets certainly an imaginary part as soon as it holds

$$-\frac{P}{2} \pm \sqrt{\left(\frac{P}{2}\right)^2 - Q} < 0. \quad (\text{B.2})$$

Implementing the abbreviations

$$P = \kappa^4 i^2 (1 + \alpha) \quad \text{and} \quad Q = \kappa^4 (1 - \kappa^4 i^4 \alpha) \quad (\text{B.3})$$

into (B.2) leads to

$$\kappa^4 > \frac{1}{i^4 \alpha}. \quad (\text{B.4})$$

After applying the relations

$$\kappa^4 = \omega^2 \frac{\rho A}{E I}, \quad i^2 = \frac{I}{A}, \quad \alpha = \frac{E A}{G A_s}, \quad A_s = \kappa A, \quad (\text{B.5})$$

and simplifying, the equation (B.4) can be rearranged to

$$\omega^2 > \frac{G A_s}{\rho I} \quad \rightarrow \quad \omega^c > \sqrt{\frac{G A_s}{\rho I}}. \quad (\text{B.6})$$

With ω^c being the critical value for ω where a change in number ranges occur. Hence, after crossing the critical frequency ω^c it holds for a fact that the solution for all four eigenvalues λ_{1-4} possess an imaginary part. Therefore, the behaviour of the eigenvalues were studied for the three frequency ranges

- I: $0 < \omega < \omega^c$,
 II: $\omega = \omega^c$,
 III: $\omega^c < \omega$.

A test example of eigenvalues for a single-span beam were computed and plotted for all three frequency domains I, II and III. Their real and imaginary parts can be studied in Figures B.1 to B.3.

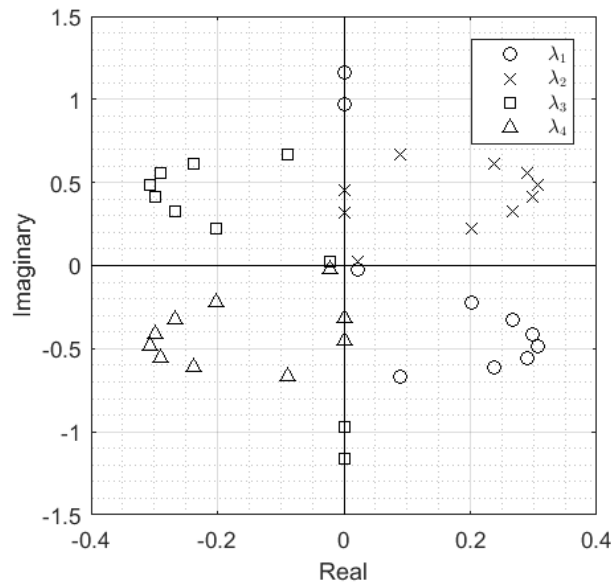


Figure B.1.: The eigenvalues for the frequency range I.

For the first frequency interval I: $0 < \omega < \omega^c$ all four eigenvalues have the form $a + ib$ consisting of a real part a and an imaginary part ib , see Figure B.1. The closer the observed frequency approaches the critical frequency, the value range for λ_{1-4} converge to the axis.

When ω reaches the exact value ω^c , the total set of four eigenvalues can be distinguished as λ_2 and λ_4 being absolutely in the real domain, and λ_1 , λ_3 dropping their real part and remaining entirely in the imaginary domain, see Figure B.2. After ω passed the critical frequency, λ_1 and λ_3 remain completely in the imaginary space, whereas λ_2 and λ_4 gain the imaginary part to end up with the form $a + ib$ again.

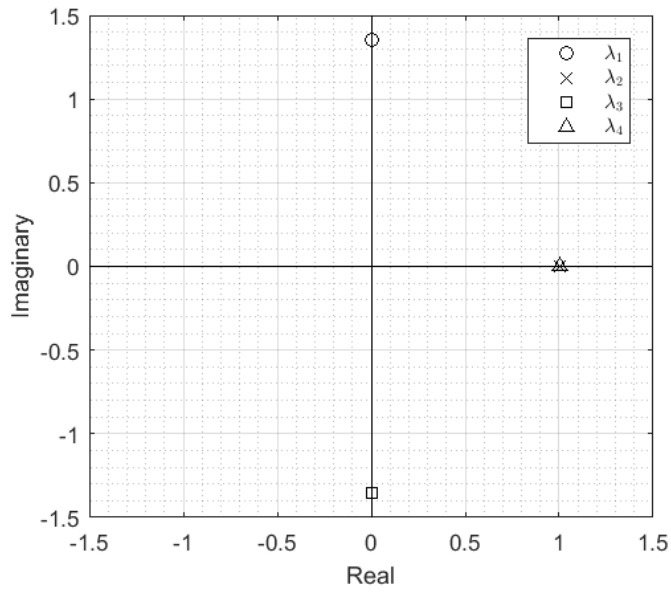


Figure B.2.: The eigenvalues for the frequency range II.

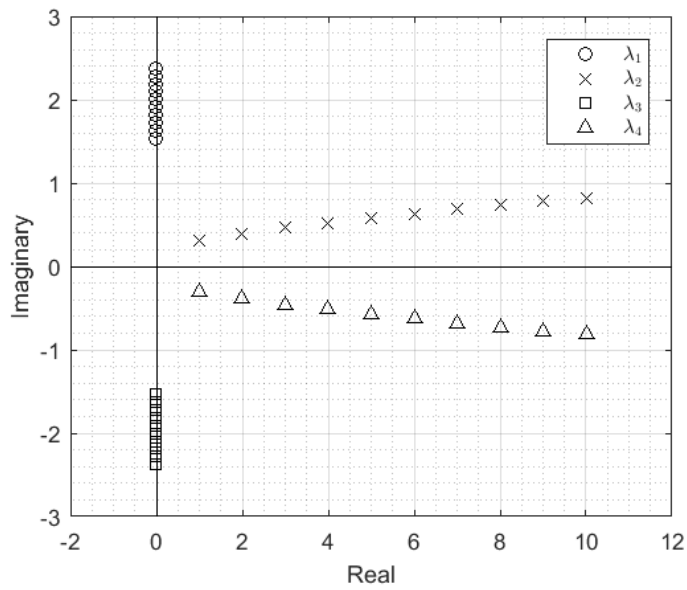


Figure B.3.: The eigenvalues for the frequency range III.

REFERENCES

- [1] W. Schiehlen and P. Eberhard, *Technische Dynamik*. Springer, 2017.
- [2] D. Dinkler, *Einführung in die Strukturodynamik*. Springer, 2017.
- [3] M. Klanner and K. Ellermann, “Steady-state harmonic vibrations of multiple-stepped Euler-Bernoulli beams under arbitrarily distributed loads carrying any number of concentrated elements,” tech. rep., Graz University of Technology, Institute of Mechanics, Kopernikusgasse 24/IV, 8010 Graz, Austria, 1 2020.
- [4] R. Gasch and K. Knothe, *Strukturodynamik-Band 2: Kontinua und ihre Diskretisierung*. Springer Verlag, Berlin, 1989.
- [5] S. Farghaly and T. El-Sayed, “Exact free vibration of multi-step Timoshenko beam system with several attachments,” *Mechanical Systems and Signal Processing*, vol. 72, pp. 525–546, 2016.
- [6] Y. Yesilce and O. Demirdag, “Effect of axial force on free vibration of Timoshenko multi-span beam carrying multiple spring-mass systems,” *International Journal of Mechanical Sciences*, vol. 50, no. 6, pp. 995–1003, 2008.
- [7] H.-Y. Lin, “On the natural frequencies and mode shapes of a multispan Timoshenko beam carrying a number of various concentrated elements,” *Journal of Sound and Vibration*, vol. 319, no. 1-2, pp. 593–605, 2009.
- [8] J.-R. Wang, T.-L. Liu, and D.-W. Chen, “Free vibration analysis of a Timoshenko beam carrying multiple spring-mass systems with the effects of shear deformation and rotary inertia,” *Structural Engineering and Mechanics*, vol. 26, no. 1, pp. 1–14, 2007.
- [9] D.-W. Chen, “The exact solutions for the natural frequencies and mode shapes of non-uniform beams carrying multiple various concentrated elements,” *Structural Engineering and Mechanics*, vol. 16, no. 2, pp. 153–176, 2003.
- [10] C. N. Bapat and C. Bapat, “Natural frequencies of a beam with non-classical boundary conditions and concentrated masses,” *Journal of sound and vibration*, vol. 112, no. 1, pp. 177–182, 1987.
- [11] J.-S. Wu and C.-T. Chen, “A lumped-mass TMM for free vibration analysis of a multi-step Timoshenko beam carrying eccentric lumped masses with rotary inertias,” *Journal of Sound and Vibration*, vol. 301, no. 3-5, pp. 878–897, 2007.

-
- [12] J.-S. Wu and C.-T. Chen, "A continuous-mass TMM for free vibration analysis of a non-uniform beam with various boundary conditions and carrying multiple concentrated elements," *Journal of sound and vibration*, vol. 311, no. 3-5, pp. 1420–1430, 2008.
- [13] J.-S. Wu and B.-H. Chang, "Free vibration of axial-loaded multi-step Timoshenko beam carrying arbitrary concentrated elements using continuous-mass transfer matrix method," *European Journal of Mechanics-A/Solids*, vol. 38, pp. 20–37, 2013.
- [14] M. Boianuiu, V. Ceausu, and C. D. Untaroiu, "A transfer matrix method for free vibration analysis of Euler-Bernoulli beams with variable cross section," *Journal of Vibration and Control*, vol. 22, no. 11, pp. 2591–2602, 2016.
- [15] S.-W. Hong and J.-W. Kim, "Modal analysis of multi-span Timoshenko beams connected or supported by resilient joints with damping," *Journal of sound and vibration*, vol. 227, no. 4, pp. 787–806, 1999.
- [16] J. Banerjee, "Free vibration of beams carrying spring-mass systems- a dynamic stiffness approach," *Computers & structures*, vol. 104, pp. 21–26, 2012.
- [17] S. Kukla, "The Green function method in frequency analysis of a beam with intermediate elastic supports," *Journal of Sound Vibration*, vol. 149, no. 1, pp. 154–159, 1991.
- [18] S. Kukla and B. Posiadala, "Free vibrations of beams with elastically mounted masses," *Journal of sound and vibration*, vol. 175, no. 4, pp. 557–564, 1994.
- [19] S. Kukla, "Application of Green functions in frequency analysis of Timoshenko beams with oscillators," *Journal of Sound and Vibration*, vol. 205, no. 3, pp. 355–363, 1997.
- [20] S. Kukla and I. Zamojska, "Frequency analysis of axially loaded stepped beams by Green's function method," *Journal of Sound and Vibration*, vol. 300, no. 3-5, pp. 1034–1041, 2007.
- [21] G. Š. Rončević, B. Rončević, A. Skoblar, and R. Žigulić, "Closed form solutions for frequency equation and mode shapes of elastically supported Euler-Bernoulli beams," *Journal of Sound and Vibration*, vol. 457, pp. 118–138, 2019.
- [22] S. Caddemi and I. Calio, "Exact closed-form solution for the vibration modes of the Euler-Bernoulli beam with multiple open cracks," *Journal of Sound and Vibration*, vol. 327, no. 3-5, pp. 473–489, 2009.
- [23] G. Failla, "On the dynamics of viscoelastic discontinuous beams," *Mechanics Research Communications*, vol. 60, pp. 52–63, 2014.

-
- [24] G. Failla, “An exact modal analysis approach to vibration analysis of structures with mass-spring subsystems and rotational joints,” *Journal of Sound and Vibration*, vol. 438, pp. 191–219, 2019.
- [25] S. S. Rao, *Vibration of continuous systems*, vol. 464. Wiley Online Library, 2007.
- [26] L. Majkut, “Free and forced vibrations of Timoshenko beams described by single difference equation,” *Journal of Theoretical and Applied Mechanics*, vol. 47, no. 1, pp. 193–210, 2009.
- [27] D. Pölz, M. H. Gfrerer, and M. Schanz, “Wave propagation in elastic trusses: An approach via retarded potentials,” *Wave Motion*, vol. 87, pp. 37–57, 2019.
- [28] T. Krapfenbauer, “Bautabellen, 19,” *Ausgabe, Wien: Verlag Jugend & Volk GmbH*, 2013.
- [29] Z. Björnstjerne, “Kleiner Ausflug in die Balkentheorie-die durchbiegung ω als zentrales element,” 2016.
- [30] N. Van Rensburg and A. Van der Merwe, “Natural frequencies and modes of a Timoshenko beam,” *Wave motion*, vol. 44, no. 1, pp. 58–69, 2006.
- [31] J. Damerau, “Untersuchung der dynamischen Eigenschaften von Balken mit fraktionalen Stoffgesetzen,” 2008.