Andreas H. Reigstad
Johannes L. Sandnes

Numerical modelling of damping for timber structures in Abaqus

June 2020

NTNU
Norwegian University of
Science and Technology
Faculty of Engineering
Department of Structural Engineering

Master's thesis

2020

Andreas H. Reigstad, Johannes L. Sandnes

Master's thesis

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Department of Structural Engineering
Faculty of Engineering
NTNU – Norwegian University of Science and Technology

ACCESSIBILITY

OPEN

MASTER THESIS 2020

SUBJECT AREA:	DATE:	NO. OF PAGES:
Timber structures, Dynamics	08.06.2020	9 + 101 + 26

TITLE:

Numerical modelling of damping for timber structures in Abaqus

Numerisk modellering av demping for trekonstruksjoner i Abaqus

BY:

Andreas H. Reigstad

Johannes L. Sandnes

SUMMARY:
Damping is a property that influence the dynamic response of timber structures. However, representing real-world damping numerically is a challenging task. In this thesis, investigations of numerical modelling of damping for timber structures have been performed by use of the commercial FE-software Abaqus by Simulia. Possibilities, limitations and usefulness of different damping models have been evaluated.

Simply supported 3D beams and rigid 2D frames have been modelled with different geometrical configurations, different elements and different mathematical damping models. These include material and global Rayleigh damping, as well as structural damping. The elements include Euler-Bernoulli and Timoshenko beam elements, a shell element, a 3D solid (volume) element and a plane stress element. A major intention was to investigate the differences between non-shear-flexible and shear-flexible elements.

Specifying material damping through the material properties module allowed for assigning different damping levels to different parts of the numerical models. Combinations of material Rayleigh damping and material structural damping have been tested. Models with global Rayleigh damping have also been assessed.

In some cases, both material and global damping deviated from the expected with respect to the mathematical formulation. Deviations were related to the choice of software architecture and element types, for which underlying causes have been investigated, although they have not been fully revealed. For the shear-flexible beam elements, implementation of material Rayleigh damping was found to underestimate damping relative to the mathematical Rayleigh damping formulation. Further investigations indicated that possible causes are related to the formulation of the Timoshenko elements and/or the establishment of the stiffness matrix. Shear effects were also considered to have an influence on deviations. Results also points towards eigenfrequencies being underestimated for high (thick) beams modelled with shell and 3D solid elements.

RESPONSIBLE TEACHER: Kjell Arne Malo

SUPERVISOR(S): Kjell Arne Malo

CARRIED OUT AT: Department of Structural Engineering, NTNU

Preface

This master thesis is the conclusion of a five-year master's program for civil engineering at NTNU. It has been carried out at the Department of Structural Engineering in the spring semester of 2020 by Andreas H. Reigstad and Johannes L. Sandnes.

Writing a master's thesis proved early on to be a new challenge for two experienced students. However, the process has been both educational and developing for both. Many thanks to employees at the Department of Structural Engineering at NTNU whom we might have bothered more than once – no one mentioned, no one forgotten. A couple of names we would still like to mention: Thanks to Øyvind W. Petersen for his contributions with code and scripts, as well as "software support" in "times of need". Special thanks to our supervisor, Professor Kjell Arne Malo, for good advice along the process – we do hope that our work will prove useful.

A special thanks to our families for encouragement and support throughout our studies – it has been invaluable.

Cheers to our fellow students for an increased interest in the field of civil engineering, a great atmosphere for learning and, last but not least, for making our time as students memorable.

Finally – thanks to both of us – for sticking out with each other through countless hours of noses stuck into an even less countable number of finite element models.

Abstract

Damping is a property that influence the dynamic response of timber structures. However, representing real-world damping numerically is a challenging task. In this thesis, investigations of numerical modelling of damping for timber structures have been performed by use of the commercial FE-software Abaqus by Simulia. Possibilities, limitations and usefulness of different damping models have been evaluated.

Simply supported 3D beams and rigid 2D frames have been modelled with different geometrical configurations, different elements and different mathematical damping models. These include material and global Rayleigh damping, as well as structural damping. The elements include Euler-Bernoulli and Timoshenko beam elements, a shell element, a 3D solid (volume) element and a plane stress element. A major intention was to investigate the differences between non-shear-flexible and shear-flexible elements.

Specifying material damping through the material properties module allowed for assigning different damping levels to different parts of the numerical models. Combinations of material Rayleigh damping and material structural damping have been tested. Models with global Rayleigh damping have also been assessed.

In some cases, both material and global damping deviated from the expected with respect to the mathematical formulation. Deviations were related to the choice of software architecture and element types, for which underlying causes have been investigated, although they have not been fully revealed. For the shear-flexible beam elements, implementation of material Rayleigh damping was found to underestimate damping relative to the mathematical Rayleigh damping formulation. Further investigations indicated that possible causes are related to the formulation of the Timoshenko elements and/or the establishment of the stiffness matrix. Shear effects were also considered to have an influence on deviations. Results also points towards eigenfrequencies being underestimated for high (thick) beams modelled with shell and 3D solid elements.

Sammendrag

Demping er en egenskap som påvirker den dynamiske responsen til trekonstruksjoner. Det er imidlertid en vanskelig oppgave å representere virkelig demping numerisk. I denne oppgaven er numerisk modellering av demping for trekonstruksjoner undersøkt ved bruk av den kommersielle FE-programvaren Abaqus fra Simulia. Muligheter, begrensninger og egnethet av ulike dempingsmodeller har blitt vurdert.

Fritt opplagte 3D-bjelker og stive 2D-rammer har blitt modellert i ulike geometriske varianter, med ulike elementer og ulike matematiske dempingsmodeller. Disse inkluderer material og global Rayleigh-demping, i tillegg til konstruksjonsdemping (eng: «Structural damping»). Elementene inkluderer Euler-Bernoulli og Timoshenko bjelkeelement, et skallelement, et volumelement og et plant spenningselement. En viktig intensjon var å undersøke forskjellen mellom elementer som er skjærfleksible og ikke-skjærfleksible.

Spesifisering av materialdemping gjennom modulen for materialegenskaper gjør det mulig å tilordne ulike nivå av demping til ulike deler av de numeriske modellene. Kombinasjoner av material Rayleigh-damping og material konstruksjonsdemping (eng: «Structural damping») har blitt testet. Modeller med global Rayleigh-demping har også blitt undersøkt.

I noen tilfeller avviker både material og global demping fra det forventede ut ifra den matematiske formuleringen. Avvik var relatert til valg av programvarearkitektur og elementtyper, hvor de underliggende årsakene har blitt undersøkt, skjønt ikke fullt ut avdekket. For de skjærfleksible bjelkeelementene førte implementeringen av material Rayleigh-demping til en underestimering av dempingen i forhold til den matematiske formuleringen av Rayleigh-demping. Videre undersøkelser indikerte at mulige årsaker er relatert til formuleringen av Timoshenko-elementene og/eller etableringen av stivhetsmatrisen. Skjæreffekter ble også vurdert til å ha en påvirkning på avvikene. Resultatene peker også mot at egenfrekvensene blir underestimert for høye (tykke) bjelker modellert med skall- og volumelementer.

1 Table of contents

1 Introduction

1.1 Background

When designing tall timber buildings there are issues that need to be considered. Firstly, it is difficult to achieve a load-bearing system that is stiff enough. Secondly, timber is a light material, having relatively much lower specific weight than other structural materials. This is a challenge because a tall and light structure will suffer more from wind-loads compared to heavier structures. DynaTTB is a project that aims to predict the dynamic response of tall timber structures [1]. Although this thesis is an independent work, it can be tied to this project as it can contribute with knowledge on damping in numerical models.

A work that has provided useful insight in the academic field of damping of timber structures, is that of Nathalie Labonnote, performed during her PhD-studies at the Department of Structural Engineering at NTNU [2]. It has laid a good foundation for further investigations.

An important property that influences the dynamic response of a tall timber building is the damping of the structure. However, representing physical damping properties in a numerical model is challenging in several aspects. Firstly, damping is a property that is not easily accurately measured, especially with respect to isolating the different damping contributions in a system and determining the portion of the total damping that each of these contribute with [3]. Which contributions that originate from material properties, and which contributions that come from structural intercomponental effects in structural assemblies, such as contact and friction, are questions that are difficult but favourable to answer prior to numerical modelling of damping.

For different damping contributions to be assigned to a numerical model in an FE-software, they must be represented mathematically. Different damping types and typical mathematical formulations of damping are presented in section 2.2. Preferably, the mathematical damping formulations should enable for the different contributions to sum up to the total system damping, so that real-world damping behaviour is adequately represented. A relevant issue is then that the different damping models are approximations, and that they to different degrees have limitations with respect to which domains of physical damping behaviour they are suitable to represent.

However, as a means towards representing damping numerically, and trying to cope with these issues, an appropriate software should be chosen. For the scope of this study, Abaqus (by Simulia) was recommended due to the many opportunities that the software provides. Although Abaqus is a software renowned for its complexity and rather tedious modelling procedures, it is also a tool that provides applicability to a wide range of engineering problems; this is particularly also the case for dynamic problems – Abaqus covers a large amount of dynamic analysis procedures. The opportunity of detailed

modelling was also considered an advantage with respect to the complexity of the problem of investigation, including both a complex physical phenomenon and a complex material.

In addition, Abaqus is an exceptionally well documented software, the documentation containing detailed information on nearly every aspect of the software. Thus, it should also be easier for the user to validate the models and compare towards analytical models when such verifications are desired. When it comes to modelling opportunities, Abaqus is thus a very complete tool and the choice should therefore be adequate.

1.2 Objective

The overall objective of this thesis is to accurately represent damping in an FE-software such that the dynamic response of a timber structure can be simulated. To achieve this, the present objectives are to get a better understanding of how damping is implemented in Abaqus, the consequences of choice of numerical damping models, and further, to investigate how structural properties influence the resulting damping.

1.3 Scope of work

This thesis continues the work done in a preliminary study performed by the authors. Initially, the options available in Abaqus are presented. Further, investigations of these have been performed. A comprehensive set of beam models with Rayleigh damping assigned have been analysed by Abaqus simulations run through a secondary software, iSight.

The work on beam models were extended to frames, and different damping models, structural damping, material damping and global damping, were investigated.

2 Theory

The basis in the field of structural dynamics is the equation of motion with associated basic definitions. These are assumed known and are only listed for reference and clarification of symbols: [4]

$$\mathbf{M}\ddot{u} + \mathbf{C}\dot{u} + \mathbf{K}u = \mathbf{p}(t) \tag{2-1}$$

$$\omega = \sqrt{\frac{K}{M}} \tag{2-2}$$

$$\zeta = \frac{c}{2M\omega} \tag{2-3}$$

$$\omega_D = \omega\sqrt{1-\zeta^2} \tag{2-4}$$

Here $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ the damping matrix, $\mathbf{K}$ the stiffness matrix, $\mathbf{p}(t)$ the external loading, u the displacement, $\dot{u}$ the velocity, $\ddot{u}$ the acceleration, ω the eigenfrequency, ζ the damping ratio and ω_D the damped eigenfrequency.

2.1 Modal analysis

Modal analysis is a classical way of solving linear dynamic multi-degree-of-freedom systems. When a system is linear it can be split up in a set of equations that are uncoupled and can therefore be solved independently. This is a very nice way of solving dynamic problems as the computational cost is low. Nevertheless, it is not always applicable and other methods might be required.

2.1.1 The eigenvalue problem

The eigenvalue problem (EVP) is a way to determine the eigenmodes and the eigenfrequencies of a system. The eigenfrequency is a frequency that the system naturally will vibrate in without any external influence, and the accompanying eigenmode determines the shape of this vibration. Once a system is described by the equation of motion, the EVP can be established. For relevant structural cases the amount of damping is usually so small that it can safely be neglected when considering the eigenmodes and eigenfrequencies. This leads to the real eigenvalue problem, which usually is just referred to as the eigenvalue problem: [4]

$$(\mathbf{K} - \omega_n^2\mathbf{M})\boldsymbol{\phi}_\mathbf{n} = \mathbf{0} \tag{2-5}$$

Here, $\boldsymbol{\phi}_\mathbf{n}$ is the eigenmode number n. The mathematics behind this is explained in many textbooks and will not be repeated here. In short, by setting the determinant of the parenthesis part in Equation (2-5) equal to zero, a set of solutions $\boldsymbol{\omega}^\mathbf{2}$ can be found. The matching eigenmode is found by solving Equation (2-5) with the now known eigenfrequency.

2.1.2 The complex eigenvalue problem

Similar as the eigenfrequencies are found from the real EVP, the damped eigenfrequencies (complex eigenfrequencies) are found from the complex eigenvalue problem, which, contrary to the EVP, take damping into account. When damping levels are small the choice to neglect damping is no problem, but this is not always the case. From the complex EVP the more accurate complex eigenfrequencies can be calculated from the equation: [4]

$$(\lambda^2\mathbf{M} + \lambda\mathbf{C} + \mathbf{K})\boldsymbol{\phi} = 0 \tag{2-6}$$

Here, λ is the eigenvalue. These eigenvalues will be found as complex-conjugate pairs $(\lambda_n, \bar{\lambda}_n)$. With $|\lambda_n| = \omega_n$ the damping ratio can be found from this:

$$\zeta_n = -\frac{Re(\lambda_n)}{|\lambda_n|} \tag{2-7}$$

2.1.3 The mode superposition method and orthogonality

The most common method used to solve both the undamped and the damped free or forced vibration problem of dynamic multi-degree-of-freedom systems, is the mode superposition method (note that this method applies only to linear (uncoupled) systems) [5]. The displacements are expressed in terms of normal coordinates – also referred to as modal coordinates or (modal) generalized displacements [3] – $\mathbf{y}$, so that $\mathbf{u} = \boldsymbol{\Phi}\mathbf{y}$, where $\boldsymbol{\Phi}$ is the mode shape matrix containing the undamped eigenvectors, obtained from solving the eigenvalue problem. Utilizing this relation, a normal coordinate transformation (modal transformation) of the equation of motion is performed. For a damped system in free vibration, the modal transformed equation of motion is [5]:

$$\boldsymbol{\phi}^{\mathbf{T}}\mathbf{M}\boldsymbol{\phi}\ddot{\mathbf{y}} + \boldsymbol{\phi}^{\mathbf{T}}\mathbf{C}\boldsymbol{\phi}\dot{\mathbf{y}} + \boldsymbol{\phi}^{\mathbf{T}}\mathbf{K}\boldsymbol{\phi}\mathbf{y} = \mathbf{0} \tag{2-8}$$

where $\mathbf{M}$ and $\mathbf{K}$ are the mass matrix and the stiffness matrix respectively.

Due to the orthogonality property of the mode shape matrix (undamped eigenvectors), both $\mathbf{M}$ and $\mathbf{K}$ become diagonal in this transformation. Conversely, a matrix that becomes diagonal when the undamped eigenvectors are used for the modal transformation, is said to satisfy orthogonality conditions (derivations of these can be found in textbooks covering the topic, such as *Dynamics of structures* by Humar [5]) [5].

Thus, if the damping matrix, $\mathbf{C}$, also becomes diagonal when the undamped eigenvectors are used for the modal transformation, $\mathbf{C}$ also satisfies damping orthogonality. Common terminology is that $\mathbf{C}$ is diagonalized, or orthogonalized, under modal transformation. Additionally, when $\mathbf{C}$ satisfies orthogonality, damping has the property of being proportional. That damping is proportional means that $\mathbf{C}$ is proportional to either or

both the mass matrix, **M**, and the stiffness matrix, **K**. Conversely, when the damping matrix is constructed so that it is proportional, it satisfies damping orthogonality conditions [5].

When all matrices in the modal transformed system are diagonal matrices, it means that the system is uncoupled, or in other words, the system consist of only linear equations, and may be solved directly for the normal coordinates, **y**. Physical displacements, **u**, are then found by superposition through the relation between physical and normal coordinates.

2.1.4 Analytical eigenfrequencies for a beam

The beam equation from Euler-Bernoulli beam theory yield an analytical expression for the eigenfrequencies related to the different vibration shapes, or mode shapes, of a beam. With the boundary conditions of a simply supported beam, the expression is as follows: [6]

$$\omega_n = n^2\pi^2\sqrt{\frac{EI}{\rho A l^4}} \tag{2-9}$$

In Equation (2-9) ω_n is the eigenfrequency, n the number of the mode, E the elastic modulus, I the second moment of inertia, ρ the density per unit length, A the cross-sectional area and l the span length.

2.2 Damping in engineering structures - terminology

Damping of a physical system is, in short, energy losses or, in other terms, energy dissipation – meaning redistribution of energy to surroundings or transformation of mechanical energy into other nonrecoverable forms of energy, typically heat – that reduces the motion of the system [3]. Numerous terms exist for different damping types and phenomena of energy losses in mechanical systems. This can lead to confusion, since different terms sometimes refer to the same physical phenomenon. Damping phenomena that are included by specific terminology may also seem to differ in the literature. A review on terminology therefore seems appropriate.

One way to classify damping is to categorize according to the sources from which damping occurs or originate (or is assumed to originate). DeSilva refer to three main classes of damping sources in mechanical systems: internal/material damping, structural damping, and fluid damping [3]. By this categorization, internal damping refers to energy dissipation within the material itself, therefore also, and more commonly, referred to as material damping. DeSilva presents two types of mathematical models for representation of this damping source: viscoelastic models and hysteretic models [3]. Further, structural damping is damping that occurs in structural assemblies

due to plasticity, dry friction (Coulomb damping), contact and interaction between structural components, typically in joints, connections, supports and other contact surfaces and interfaces. Fluid damping refers to drag forces and friction that occurs between a structural component and a fluid (like water, or air) due to the viscosity of the fluid [3].

Fluid-structure interaction for ground-based structures (excluding offshore structures or structures typically investigated in fluid dynamics), like timber buildings, is typically related to wind. Damping mechanisms resulting from wind are quite complex phenomenon. Fluid damping is not subject to investigation in this work.

The way DeSilva classify damping sources could seem to be practical for structural engineering purposes. The advantage would be clearly specified domains (materials, connections, interfaces etc.) for which damping is to be assigned to a numerical FE-model representing a structure. On the other hand, this approach may also have its shortcomings. Separating damping contributions into different damping sources is a difficult task, particularly when it comes to measurements. Using different mathematical damping models for different parts of a structure, could also be challenging with respect to compatibility between damping models: finding adequate input damping values for individual parts of a structure that give predictable output damping values for the entire structure is not straightforward.

When a classification like the above is used, it should also be noted that each damping source category may not merely be represented by one type of mathematical damping formulation, but several. Viscous damping models are for instance not only used for representing fluid damping but are used to approximately represent other damping sources as "viscous-like". The viscous damping force term, $f_D = c\dot{u}$, in the equation of motion is based on the assumption that the damping force is proportional to the velocity of a system for small velocities [5]. This kind of property is relatively easy to formulate mathematically, and thus, it should be evident that viscous damping is utilized for numerical purposes – note that when viscous damping is formulated mathematically or modelled in finite element software it is generally related to the viscous term in the equation of motion. In the end, the goal is to simulate damping forces that reduce the motion of a system – then the means towards accomplishing this are perhaps subordinate when better methods are absent.

Another way of classifying damping is simply to group into mathematical damping model categories based on which damping phenomena they intend to represent. A representative example is the way Humar, in his textbook *Dynamics of structures,* classify damping according to the nature (behaviour) of the damping mechanisms, and properties of the corresponding mathematical formulations [5]. According to Humar, the most important damping types in engineering structures are viscous damping, dry friction effects called Coulomb friction, and hysteretic or structural damping [5].

From this categorization, it becomes evident that terminology is used differently among professionals in the academic field. Coulomb damping would classify as structural damping according to DeSilva, while the term structural damping on several occasions, such as in Humar's definition, is used as an equivalent term to hysteretic damping represented by complex stiffness [5]. In the latter case, in which structural damping and hysteretic damping are used synonymously, both terms refer to internal frictional damping mechanisms – however, "internal" in the sense of internal damping effects within the material itself, not (internal) structural damping within a structural assembly due to relative motion between structural components. The inconsistent use of terminology is apparently substantiated. Anyhow, as will be explained in section 2.2.5, the term "hysteretic" is in fact representative for all types of damping [3]. Arguments could therefore be made that regardless of the use of the terms hysteretic damping and structural damping, either ways might be appropriate, as long as the context of use is adequately explained [3].

Still, it is still quite evident that any universal classification of damping types and sources seem to be non-existent – there is great variability in use of terminology, and lack of consistency in the academic field. Trying to follow a consistent classification is thus not easy to comply, although awareness of this issue is advantageous.

The common approach to modelling of damping in finite element software, is that the software include different mathematical damping models which may be assigned either to a material definition, directly onto the entire global model, and/or modal damping may be specified typically for a range of vibration modes or for a range of frequencies. Damper elements such as viscous dashpots, connectors, and springs with complex stiffness are also common tools. Technically, modelling of damping in FE-software is thus a question of which mathematical damping models to assign to the numerical models and how these may be combined to approximately represent real-world damping behaviour; it is a question of controlling the input damping and predicting the output damping levels of a numerical model. An FE-software provide a set of tools and does not ask whether the use of these tools is appropriate – this is left for the user to consider.

From the above discussion, it should be clear that the important when working in the academic field considering damping of structures – particularly when attempting to model damping numerically – is to have a basic understanding of the applicability and limitations of existing damping models, so that these may be used and combined in ways that are applicable to common problems, and efficiently solve these. The theory behind damping types that are used in numerical models in this work, are presented in the following sections.

2.2.1 Viscous damping

A damping force working on a dynamic system will necessarily work in the opposite direction to the motion of the system, i.e. in the opposite direction of the velocity [5]. This may be expressed in the equation of motion as a damping force, f_D – usually referred to as the viscous damping force – that is proportional to velocity so that $f_D = c\dot{u}$, where c is a damping constant (proportionality constant) and $\dot{u}$ is the velocity [5]. Different mathematical damping formulations use different approaches to express the damping constant, c.

The above way of expressing the damping force, is the common definition of viscous damping introduced in the equation of motion. Experiments show that for small velocities the damping force that arise from interaction between a viscous fluid and a structural component is proportional to the relative velocity between fluid and component [5]. Although viscous damping in this sense is a specific type of damping phenomena and, strictly speaking, applies primarily to represent actual viscous or viscous-like system behaviour, the "simple" way in which viscous damping can be modelled numerically makes it an efficient tool that may be used to represent other damping phenomena (approximately) as well.

For a multi-degree-of-freedom system, the equation of motion turns into a system of equations. It follows that a damping matrix, **C**, of multiple constants must be specified.

As previously discussed, when damping is proportional, the damping matrix satisfies orthogonality, meaning that the system of dynamic equilibrium equations becomes linearized (uncoupled) in a modal transformation, and the dynamic equilibrium equations may be solved by modal superposition. One widely used and efficient method to express proportional viscous damping numerically, is Rayleigh damping. Rayleigh damping is a formulation in which the damping matrix, **C**, to a system, is defined as a linear combination of the mass and stiffness matrices, **M** and **K,** to the system. This is done by specifying two proportionality constants, the Rayleigh damping coefficients, α_R and β_R, respectively for mass- and stiffness proportional damping. Rayleigh damping is thus expressed as [5]:

$$\mathbf{C} = \alpha_R \mathbf{M} + \beta_R \mathbf{K} \tag{2-10}$$

Since Rayleigh damping satisfies orthogonality one may decompose the damping matrix to modal form, for which damping, C_i, of vibration mode "i", can be expressed as:

$$C_i = \alpha_R M_i + \beta_R K_i \tag{2-11}$$

By substituting the relation $C_i = 2\zeta_i \omega_i M_i$ for the modal damping constant and rearranging, the expression for the modal (critical) damping ratio, ζ_i, for mode i reads [4]:

$$\zeta_i = \frac{\alpha_R}{2\omega_i} + \frac{\beta_R \omega_i}{2} \tag{2-12}$$

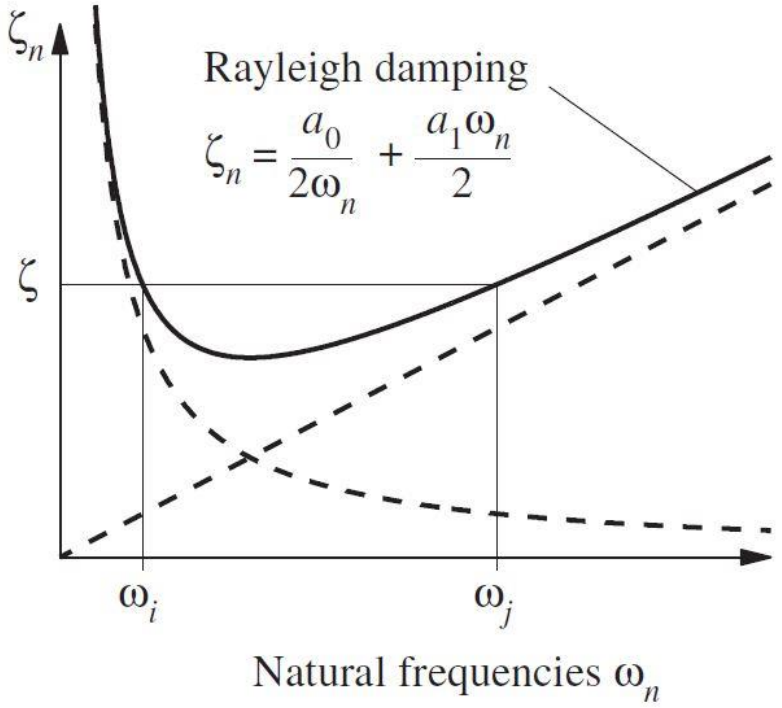

Figure 2.1: The Rayleigh damping curve [4].

For specified values of the Rayleigh damping coefficients, Equation (2-12) may be represented graphically as the Rayleigh damping curve, for which the damping ratio is plotted for different frequencies; this is displayed in Figure 2.1. In the figure, a_0 corresponds to α_R and a_1 corresponds to β_R.

2.2.2 Modal damping

Modal damping may be introduced by any method that allows for specification of damping ratios for the vibration modes. The idea is that any displacement can be represented by a set of mode shapes, or modes. In an FE-software, one can measure the damping in each of these modes or assign damping to each mode. Rayleigh-damping as specified by Equation (2-12) above is an example of a common modal damping method [5].

2.2.3 Nonproportional damping – Nonlinear systems

If the damping matrix is not diagonalized in the modal transformation that uses the undamped mode shapes, the damping matrix does not satisfy orthogonality, i.e. damping

is not proportional, and the system equations are coupled. In other words, nondiagonal damping means that the system of equations are nonlinear, and, followingly linear methods like the modal superposition method cannot be used to solve the problem [5]. If discrete damper elements like dashpots (idealized viscous damping elements) are modelled into a structural system, the resulting damping matrix will in general be nonlinear [3]. Nonlinear equations must of course also be used if nonlinear response is being studied.

Nonlinear problems must be solved by use of a numerical (iterative) method, which involves direct integration of the equations of motion. Direct integration procedures require that the damping matrix is specified [5].

2.2.4 Solving nonlinear systems – Implicit time integration and numerical damping

Numerical damping is a mathematical phenomenon (property) that every user of direct integration procedures – in FE-software and in general – should be aware of. This is damping, or artificial energy loss, that occurs during execution of direct time integration of a numerical scheme. For nonlinear analyses the operators are required to be implicit, which means that the solution at the next time step is calculated not only from solutions at previous time steps, but also from approximate solutions at this next time step. Consequently, implicit schemes form a set of nonlinear equations that must be solved at each time step [7]. It could be noted that implicit schemes may also be used, although they are not required, to solve linear problems.

Some numerical schemes allow for controlling the level of numerical damping occurring from integration of the scheme – such control happens through integrator parameters, which are constant coefficients that may be pre-set. The opportunity to control the level of numerical damping for respective schemes is thus dependent on whether they contain any integrator parameters, and, if so, the properties of these.

Controlling the amount of numerical damping that is introduced in a model is advantageous; for certain amounts of numerical damping, numerically, it can be a means to improve the convergence rate (or in case, to assure convergence) of the analysis, while maintaining adequate accuracy of the solution [8].

In general, evaluation of the stability, i.e. convergence criteria, of a numerical scheme is important, and size of the time increments should be set so that convergence is assured. Too large time steps may result in lesser accuracy of the response prediction [7], and typically it will increase the amount of numerical damping [8]. Choosing the time step so that the ratio between the time increment and the vibration period of the structure is less than $\frac{1}{10}$ is usually adequate [5], [7].

2.2.5 Hysteretic damping and structural damping

As known from linear elastic theory, as long as the strain level does not exceed the elastic range the stress/strain-curve (or force/displacement-curve) of an ideal linear elastic material exposed to cyclic loading will be a straight line, indicating that no energy is lost during the load cycle [5]. In reality, such a material or mechanical system does not exist – straining of materials or structural systems will always result in some energy losses, meaning energy dissipation or conversion of energy into other forms of energy (typically heat) [5]. A realistic stress/strain-curve will thus not be a straight line, but a loop, for instance as displayed in Figure 2.2. The physical phenomenon represented by such a curve is referred to as hysteresis, or, in damping terms, as hysteretic damping. In this sense, all systems that are strained have hysteretic damping that reduces or opposes the motion of the system. The amount of hysteretic damping that occurs in a material or a structural system during each load cycle, or vibration period, is quantified by the area inside the hysteresis loop (note that Figure 2.2 is only a simplified principle sketch; a hysteresis curve will neither generally nor typically be of exact elliptical shape (except in special cases)). [5]

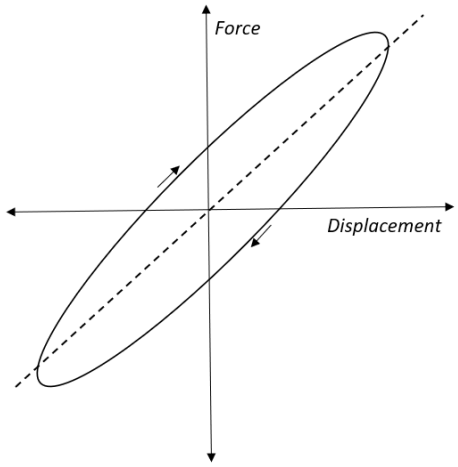

Figure 2.2: Hysteresis loop.

The term structural damping is frequently used synonymously to the term hysteretic damping, although neither terms are used consistently with respect to whether they refer to internal damping of materials or to damping mechanisms related to internal interaction between structural components. As long as the terms are interpreted in the context for which they are used, or mathematical damping formulations are clear on which damping phenomena they may represent, this should not be a major issue. Usually, it is neither a problem in a numerical sense: although terminology might refer to different damping mechanisms, the mathematical formulations that represent hysteretic and structural damping in some cases are the same.

A common method for expressing hysteretic or structural damping numerically, that applies regardless of use of terminology, is to model damping through complex stiffness [5], [8]. The starting point is the equation of motion for forced harmonic motion with hysteretic damping, which reads:

$$m\ddot{u} + f_S^t(u, \dot{u}) = p(t) \tag{2-13}$$

where $f_S^t(u, \dot{u})$ is referred to as the total spring force and $p(t)$ is the forcing term [5]. f_S^t consist of the spring force, $f_S = ku$, and the hysteretic damping force, $f_D = \frac{\eta k}{\Omega}\dot{u}$, for which k in both cases is the real valued stiffness, η is a damping factor (loss factor; a constant), and Ω is the frequency of the driving force. The equation of motion may then be written as $m\ddot{u} + \frac{\eta k}{\Omega}\dot{u} + ku = p(t)$.

Further, when the driving force is expressed as $p(t) = p_0 e^{i\Omega t}$ a solution may be on the format $u = Ue^{i\Omega t}$. Differentiating u gives the velocity, $\dot{u} = Ui\Omega e^{i\Omega t} = i\Omega u$, which, when substituted into the expression for the hysteretic damping force, f_D, expresses this on complex form:

$$f_D = i\eta k u \tag{2-14}$$

Substituting back into the equation of motion yields

$$m\ddot{u} + i\eta k u + ku = p(t) \tag{2-15}$$

which simplifies to

$$m\ddot{u} + \bar{k}u = p(t) \tag{2-16}$$

where complex stiffness is introduced (defined) as $\bar{k} = (1 + i\eta)k$ [5].

As seen from the expression, structural or hysteretic damping modelled by complex stiffness, has the property of being independent of frequency. This property distinguishes this type of damping from viscous damping models which are frequency dependent. This is an advantage, since some experiments show that damping is not dependent on frequency [5]. Structural damping may therefore be an alternative to viscous damping models.

It could be mentioned that hysteretic damping as defined here, is not adequate for representing energy losses in systems that behave nonlinearly, as is the case in occurrence of plastic deformations [5].

2.3 Logarithmic decrement method

The logarithmic decrement method is a method in which the damping ratio of an underdamped system in free vibration may be determined from experimental response measurements [3]. The response, u, at time t, of such a system is $u = \rho \cos(\omega_d t + \theta)\, e^{-\zeta\omega_n t}$, where ρ is the response amplitude, θ is the phase angle, $\omega_d = \omega_n\sqrt{1 - \zeta^2}$ is the damped frequency of vibration, ω_n is the natural frequency and ζ is the damping ratio of the system [5]. A representative plot of response as a function of time is displayed in Figure 2.3.

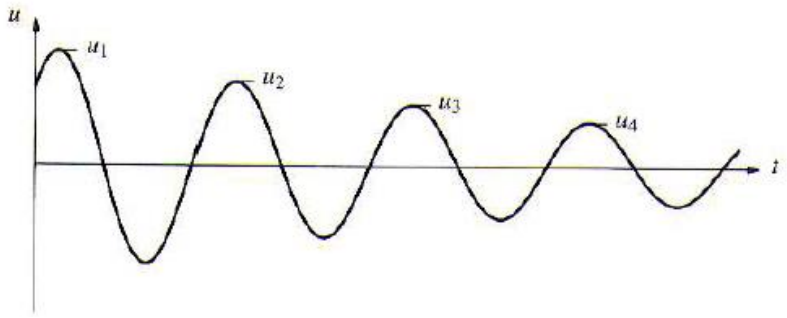

Figure 2.3: The response of a freely vibrating damped system [9]. Response peaks are marked in the plot.

Taking the ratio between two response peak values, u_n and u_{n+m}, respectively at time $t_n = nT_d$ and $t_{n+m} = (n+m)T_d$, where sub-index "m" means that t_{n+m} is m periods after t_n, and $T_d = \frac{2\pi}{\omega_d}$ is the damped vibration period, gives

$$\frac{u_n}{u_{n+m}} = \frac{\rho\cos(\omega_d t_n+\theta)e^{-\zeta\omega_n t_n}}{\rho\cos(\omega_d t_{n+m}+\theta)e^{-\zeta\omega_n t_{n+m}}}$$

which, when inserting relations for t_n, t_{n+m} and ω_d simplifies to

$$\frac{u_n}{u_{n+m}} = e^{-\zeta\omega_n(n-(n+m))T_d} = e^{\zeta\omega_n mT_d} = e^{2\pi m\zeta\frac{\omega_n}{\omega_d}} = e^{2\pi m\frac{\zeta}{\sqrt{1-\zeta^2}}}.$$

Further simplification follows by assuming low levels of damping, so that $\omega_d \approx \omega_n$, which gives $\frac{u_n}{u_{n+m}} = e^{2\pi m\zeta}$.

By taking the natural logarithm on both sides, the equation may be solved for the damping ratio:

$$\zeta = \frac{\delta}{2\pi m} \tag{2-17}$$

where $\delta = \ln\left(\frac{u_n}{u_{n+m}}\right)$ is the logarithmic decrement [9], for which u_n and u_{n+m} may be read directly from the response plot.

2.4 MAC – Modal assurance criterion

The modal assurance criterion, commonly abbreviated MAC, is a method used to compare mode shapes (eigenvectors) of dynamic systems, typically theoretical (analytical) and experimental modes are compared. MAC measures the correlation (or consistency) between modes as a value between 0 and 1, usually presented as a percentage, for which MAC = 1 indicates 100% correlation between two modes. Although the "limits" for correlation are not absolute, usually, modes are considered to be well-correlated when the MAC-value is 0.9 or higher, while a MAC-value below 0.1 indicates (strongly) uncorrelated modes [10].

In numerical terms MAC is a least square's deviation measurement method. For comparison of two mode shape vectors, $\{\boldsymbol{\psi}_\mathbf{A}\}$ and $\{\boldsymbol{\psi}_\mathbf{B}\}$, of two vibration modes, A and B, MAC may be defined as: [10]

$$MAC(A,B) = \frac{|\{\boldsymbol{\psi}_\mathbf{A}\}^{*\mathbf{T}}\{\boldsymbol{\psi}_\mathbf{B}\}|^2}{(\{\boldsymbol{\psi}_\mathbf{A}\}^{*\mathbf{T}}\{\boldsymbol{\psi}_\mathbf{A}\})(\{\boldsymbol{\psi}_\mathbf{B}\}^{*\mathbf{T}}\{\boldsymbol{\psi}_\mathbf{B}\})} \tag{2-18}$$

where "$*$" denotes the complex conjugate and "T" denotes the transpose, i.e. "*T" denotes the complex conjugate transpose (commonly referred to as the Hermitian transpose, denoted "H"). When the complex conjugate transpose is used, input mode shape vectors may be both real valued and complex – mode shape vectors then consist of both the real and the imaginary part. Either way, the resulting MAC-value is a scalar [10].

There are some limitations regarding interpretability of MAC-values that follows from the definition, and some common pitfalls that are important to be aware of. Firstly, MAC is not sensitive to scale (amplitude) differences between modes shapes. Although a MAC-value of 1 indicate 100% correlation between mode shapes, it is no guarantee that the mode shapes are numerically identical. In the case that the compared mode shapes are linearly related (linearly dependent) – meaning that they are scalar multiples of one another – according to MAC, the modes will be perfectly correlated (if other errors are not present) even though the amplitudes of the mode shapes are not equal [10].

Additionally, since MAC is based on the least square's method, large magnitude differences between degrees of freedom in the respective modes will govern the calculation. Therefore, a MAC-value will be influenced relatively more by large discrepancies than small. If, for instance, large-magnitude errors are present, they might govern the resulting MAC-value [11].

Numerical noise (random or biased) in the eigenvectors is another potential error that will influence the MAC-calculation. If present in both eigenvectors it could give well-correlated modes, even though they are in fact not physically interpretable. Mode shapes should therefore be checked for such errors [11].

It is also important to assure that an adequate selection of DOFs is used in the MAC-analysis, and that sufficiently many DOFs are included for the MAC-value to be reliable. If a limited selection of a large set of DOFs is used, a potential error may be that the selected DOFs do not represent the mode shapes they are supposed to represent. In general, the more DOFs included, the higher the accuracy of the MAC-value. A special case of MAC, called AutoMAC, is simply a MAC-calculation in which a mode shape is compared to itself [10]. This method may be used to assure that the number of DOFs included is sufficient and that the selection of DOFs is adequate, as well as for checking that the implementation of MAC is correct [10]. AutoMAC should always give a value of 1, otherwise errors might be present, and/or the DOF-selection might be inadequate.

3 Damping in Abaqus

The objective of this chapter is to give the reader an adequate review on the options for modelling of damping that is provided by Abaqus. Much of the content is based on a preliminary study performed by the authors prior to this work [12], although this review is somewhat more extensive: the sections are modified and some extra topics have been included according to the scope of this work. It should further be noted that all of the content is based on the Abaqus documentation version 6.14, particularly the *Analysis User's Guide* [8]and the *Theory Guide* [7]. The Abaqus documentation is vast, and the reader is referred to this for complementary details – here is attempted to provide only the most essential (and required) information on numerical modelling of damping.

For completeness a summary of the dynamic analysis procedures in Abaqus are presented, prior to more details regarding Abaqus damping types, methods and procedures for modelling of damping.

3.1 Software system programs

Abaqus by Simulia is a finite element software that includes several programs and environments. The relevant in the context of this study are the general-purpose program Abaqus/Standard and the interactive user-interface Abaqus/CAE [8].

3.2 Dynamic analysis procedures in Abaqus

The dynamic analysis procedures that are provided by Abaqus are divided into two main categories: direct integration procedures and modal procedures. A direct integration procedure may be used for study of both linear and nonlinear response; however, implicit direct integration is a requirement for study of nonlinear systems and response. Modal superposition procedures are computationally much more cost-effective and are therefore preferable for study of linear response systems. Additionally, within both two main categories so-called subspace-based methods are included. They may be used for both linear and nonlinear systems. However their primary advantage compared to direct integration solvers, is that they can provide time-efficient calculations of mildly nonlinear systems [8]. That the system is mildly nonlinear, involves that the mode shapes are not greatly influenced by the nonlinearities [7] , [8].

The provided direct integration procedures are [8]:

- Implicit dynamic analysis (Abaqus/Standard)
- Subspace-based explicit dynamic analysis (Abaqus/Standard)
- Explicit dynamic analysis (Abaqus/Explicit)
- Direct-solution steady-state harmonic response analysis

The available modal procedures in the software are [8]:

- Mode-based steady-state harmonic response analysis
- Subspace-based steady-state harmonic response analysis
- Mode-based transient response analysis
- Response spectrum analysis
- Random response analysis
- Complex eigenvalue extraction

In addition to the default Abaqus "traditional" software architecture, several dynamic analyses may be run with another architecture called SIM. The SIM architecture is a high-performance software architecture that can improve the performance and calculation efficiency of linear dynamic modal superposition procedures, such as steady-state dynamics and transient dynamic response analysis. [8]. The SIM architecture is activated in an initial frequency extraction step, that must be included prior to any dynamic analysis step in which SIM advantages are to be achieved. The damping projection option, "project damping operators", in which the damping parameters are projected from one step (the initial frequency extraction step) onto subsequent dynamic analysis steps, is thus only available when SIM architecture is used. Moreover, several damping types within the damping source categories also require use of SIM, as displayed in Table 3.1 Another situation in which SIM is required, is when the damping matrix is non-diagonal, i.e. have off-diagonal contributions, which in general is the case when damping is modelled by damper elements such as springs or dashpots [3], [8].

3.3 Modelling of damping in Abaqus

Contributions to the total damping of a system come from the damping sources in the system. In Abaqus there are four damping source categories available [8]:

- Material and element damping.
- Global damping.
- Modal damping.
- Damping associated with time integration (numerical damping).

The damping sources may be implemented in a model alone or in combination with one or several of the other sources, dependent on dynamic procedure and executing software architecture. Available damping sources and optional combinations of sources within the different dynamic analysis procedures, for both traditional Abaqus software architecture and the SIM architecture, are presented in Table 3.1.

Table 3.1: Available damping sources for different software architecture. [8]

Traditional Architecture	**Damping source**		
	Material and element	**Global**	**Modal**
Mode-based steady state dynamics		X	X
Subspace-based steady-state dynamics	X	X	
Transient modal dynamics		X	X
Random response analysis		X	X
Complex frequency	X	X	
Response spectrum		X	X
Direct steady-state dynamics	X	X	
SIM architecture	**Damping source**		
	Material and element	**Global**	**Modal**
Mode-based steady state dynamics	X	X	X
Subspace-based steady-state dynamics	X	X	X
Transient modal dynamics	X	X	X
Random response analysis		X	X
Complex frequency	X	X	X
Response spectrum		X	X

In material and element damping in Abaqus, material damping is specified as part of the material definition; element damping is damping represented by mechanical damping components like dashpots, springs or connectors. Global damping is specified by so-called global damping factors, that obviously applies to the entire model. When global damping is used, it is usually in addition to material damping. The advantage of modal damping is that damping is specified onto the respective vibration modes in the dynamic system, either so that the damping is different for each respective mode, or the same for all modes, depending on the mathematical damping model [8].

Damping associated with time integration, more commonly termed numerical damping, differs from the other damping sources in that it is not a representation of any physical phenomenon or behaviour of a material or dynamic system. Numerical damping is simply damping that occurs during direct time integration of a numerical scheme – note that numerical schemes are also referred to as “operators” or “integrators” in Abaqus [7], [8].

The three damping source categories, material and element, global, and modal damping, is introduced into an Abaqus FE-model through two main types of mathematical damping models: The first main damping type is velocity proportional viscous damping, the second is displacement proportional structural damping. The mathematical formulation of both the two types differ slightly depending on which damping sources

are represented. Velocity proportional viscous damping in Abaqus is mainly represented by Rayleigh-damping. Modelling of structural damping is done by use of complex stiffness. For mode-based dynamic procedures, an extra damping type option, composite damping, is available. This is a modal damping type that can be assigned to the material/element level [8].

Since Abaqus provide quite many options for modelling of damping, only those that have been investigated in detail in this work, and some that has been considered as relevant to mention, are described in the following sub-chapters. The reader is referred to the Abaqus documentation for other options and complementary details on those that are presented here.

3.3.1 Modelling of damping – Material and element damping

Material damping is modelled as a part of the material definition in the material properties module. Element damping is damping introduced through damper elements. In Abaqus, this damping may be represented by proportional viscous damping, non-proportional viscous damping, structural damping, and composite damping.

3.3.1.1 Material and element damping – proportional and nonproportional viscous damping

Material and element viscous damping is represented numerically by the viscous damping matrix, $\mathbf{D}^{\mathbf{el}}_{\mathbf{viscous}}$, which may be proportional to either or both the mass matrix and the stiffness matrix, depending on which contributions are specified. The contributions are included by defining damping in the material property definition, by modelling of mass elements and other damping elements like dashpots or connectors. In the material definition the Rayleigh-damping coefficients, α_R^{mat} and β_R^{mat}, are specified to add damping through the material properties. One type of damper elements, mass elements with corresponding mass matrix $\mathbf{m}^{\mathbf{el}}$, have damping defined similarly by specifying separate mass proportional element Rayleigh-damping coefficients, α_R^{el}. In general, α_R is specified for mass proportional damping; $\beta_R$is specified for stiffness proportional damping – normally both are specified, but it is possible to include either just mass- or stiffness proportional damping in a model. Another common type of damper elements are dashpots, which are represented by viscous damping matrices, $\mathbf{d}^{\mathbf{el}}_{\mathbf{viscous}}$, one per element. When all available options are included in the model, the resulting viscous damping matrix is defined as [8]:

$$\mathbf{D}^{\mathbf{el}}_{\mathbf{viscous}} = \sum_{el=1}^{\#\,elmnts} \int_V \alpha_R^{mat} \mathbf{N}^{\mathbf{T}}\mathbf{N}\,\rho dv + \sum_{el=1}^{\#\,elmnts} \int_V \beta_R^{mat} \mathbf{B}^{\mathbf{T}}\mathbf{D}\mathbf{B} dv + \sum_{el=1}^{\#\,elmnts} \alpha_R^{el}\,\mathbf{m}^{\mathbf{el}} + \sum_{el=1}^{\#\,elmnts} \mathbf{d}^{\mathbf{el}}_{\mathbf{viscous}} \tag{3-1}$$

In the expression, **N** are the element interpolation functions, **B** are the derivatives of **N**, which give the strains (second derivatives for curvatures in 1D-beam elements), and ρ is the material density. The damper elements in general give off-diagonal (nonproportional) contributions to the system damping matrix, i.e. making $\mathbf{D}^{\mathbf{el}}_{\mathbf{viscous}}$ non-diagonal [3]. In other words, material and element damping may give both proportional and nonproportional damping [8].

3.3.1.2 Material and element structural damping

Structural damping in Abaqus uses a hysteretic damping model defined by complex stiffness as described in section 2.2.5 [8]. This corresponds to the definition presented by Humar [5], although the Abaqus documentation states that it may also be used for modelling of damping properties that corresponds to structural damping as defined by DeSilva [13], [8].

Structural damping properties may be modelled as material and/or element structural damping in Abaqus by specifying either or both of these contributions to the structural damping matrix, $\mathbf{K}^{\mathbf{m}}_{\mathbf{s}}$, which, including contributions from both, reads [8]:

$$\mathbf{K}^{\mathbf{m}}_{\mathbf{s}} = \sum_{el=1}^{\#\ elmnts} \int_V s\mathbf{B}^{\mathbf{T}}\mathbf{D}\mathbf{B}dv + \sum_{el=1}^{\#\ elmnts} s^{el}\mathbf{k}^{\mathbf{el}} \tag{3-2}$$

where **B** and **D** is defined equally as in Equation (3-1). The material structural damping part is included by specifying the structural damping factor, "s", in the material property definition. Element structural damping comes in by modelling spring- or connector-elements, and specifying the structural damping factor, s^{el}, for the respective elements, with their associated real element stiffness matrices, $\mathbf{k}^{\mathbf{el}}$ [8].

Since $\mathbf{K}^{\mathbf{m}}_{\mathbf{s}}$ is the imaginary part of complex stiffness, it could be noted for clarity, that as it is defined, it is real-valued, since the structural damping factor, s, is the equivalent (in Abaqus-notation) to the structural damping factor η, defined in theory section 2.2.5 [7]. The corresponding in Abaqus notation to the multi-degree-of-freedom expansion of the complex stiffness $\bar{k} = (1 + i\eta)k$ defined in section 2.2.5 is thus $(\mathbf{K} + \mathrm{i}\mathbf{K}^{\mathbf{m}}_{\mathbf{s}})$, where **K** is the real elastic stiffness matrix and "i" is the imaginary number [7], [8].

3.3.1.3 Composite (modal) damping – modelled in material module

The modelling procedure of composite modal damping is different whether SIM is used or not. Abaqus differs between specifying the parameters "COMPOSITE" for the traditional architecture, and the parameter "COMPOSITE MODAL DAMPING" for SIM architecture [8]. Only the traditional architecture variant, which is the only option in Abaqus/CAE, is presented here; the reader is referred to the documentation for the SIM-alternative.

Composite modal damping is a modal damping model in which damping, through the damping fraction, ξ_α, is calculated for the respective modes, α, of a vibrating system. Although this is a modal damping type, it is included here in the material damping section, since the user input happens in the material properties module – it is done by specifying a proportion of critical damping, ξ_m, for each material or element, prior to the calculation of the composite modal damping parameters, ξ_α, performed by the software. The calculation is according to the expression [8]:

$$\xi_\alpha = \frac{1}{m_\alpha} \boldsymbol{\phi}_{\boldsymbol{\alpha}}^{\mathbf{T}} \left(\sum_m \xi_m \mathbf{M}_{\mathbf{m}} \right) \boldsymbol{\phi}_{\boldsymbol{\alpha}}, \tag{3-3}$$

where subscript α refer to the vibration mode number, subscript m refer to material number, m_α is the generalized mass, resulting from modal transformation of the system mass matrix $\mathbf{M}$, $\mathbf{M_m}$ is the material mass matrix, and $\boldsymbol{\phi}_{\boldsymbol{\alpha}}$ is the eigenvector. The principle of composite modal damping is thus that damping is calculated as a mass-weighted average for every mode [8].

3.3.2 Modelling of damping – Global damping

Modelling of global damping is not possible in Abaqus/CAE, only through specifying global damping parameters in a job-input file. Global damping differs from material damping in that it is specified for the whole model. Both global viscous damping and global structural damping have similar definitions as the corresponding material damping equivalents [8].

3.3.2.1 Global viscous damping

Global viscous damping is modelled by specifying the global Rayleigh-coefficients, α_{global} and β_{global} , respectively for mass and stiffness proportional damping contributions. The resulting global damping matrix is [8]:

$$\mathbf{D}_{\mathbf{viscous}}^{\mathbf{g}} = \alpha_{global}\mathbf{M} + \beta_{global}\mathbf{K} \tag{3-4}$$

If viscous damping is specified both on material level and global level, the contributions sum up to the system viscous damping matrix, i.e. $\mathbf{D}_{\mathbf{viscous}} = \mathbf{D}_{\mathbf{viscous}}^{\mathbf{el}} + \mathbf{D}_{\mathbf{viscous}}^{\mathbf{g}}$ [8].

3.3.2.2 Global structural damping

One parameter, the global structural damping factor s_{global} must be specified for global structural damping to be applied to an Abaqus model. The global structural damping matrix is defined as [8]:

$$\mathbf{K_s^g} = s_{global}\mathbf{K}, \tag{3-5}$$

where $\mathbf{K}$ is the real system stiffness matrix. If both global and material structural damping is specified, both contributions are added up into a combined system structural damping matrix, $\mathbf{K_s} = \mathbf{K_s^m} + \mathbf{K_s^g}$ [8].

3.3.3 Modelling of damping – Modal damping

Apart from composite modal damping, modal damping types in Abaqus include viscous modal damping and structural modal damping. The latter is not reviewed here.

3.3.3.1 Viscous modal damping

Viscous modal damping may be specified as proportional damping, similarly as for material and structural viscous damping, or the user may specify modal damping as a fraction of critical damping.

In the first case Rayleigh damping is specified separately for each mode, M, or for a frequency range. The modal Rayleigh-coefficients, α_M and β_M, are assigned in a linear perturbation step, and the modal damping matrix, c_M, for mode M, reads [8]:

$$\mathbf{c_M} = \alpha_M \mathbf{m_M} + \beta_M \mathbf{k_M} \tag{3-6}$$

where $\mathbf{m_M}$ and $\mathbf{k_M}$ are the modal mass and stiffness matrices for mode M.

Modelling viscous modal damping as a fraction of critical damping, is done by creating a valid linear perturbation step and specifying the critical damping ratio, ξ_i, for modes i or for a range of frequencies.

3.3.4 Numerical damping

As reviewed in theory section 2.2.4, numerical damping is damping related to direct time integration procedures and might be controlled by eventual integrator parameters contained by the operator. Numerical operators used for direct integration procedures in Abaqus/Standard are implicit and may therefore be used to solve both linear and nonlinear problems. The nonlinear equations formed by the numerical implicit schemes are solved at each time step by Newton's method [8].

The implicit schemes that Abaqus provide are the Hilber-Hughes-Taylor operator and the backward Euler operator [7]. User control of numerical damping occurring for the backward Euler scheme is not possible, since the operator does not have any integrator parameters. However, this scheme is intended for quasi-static dynamic analyses, which are not investigated in this work [8].

For procedures that analyse moderately and lightly damped nonlinear systems, and other systems that has some sort of nonlinearities, the default operator is the Hilber-Hughes-Taylor scheme. This operator has three integrator parameters, α, β, and γ, that allows the user to control the amount of numerical damping. Both the β- and the γ-parameter is dependent on α, so that user specification of the latter, automatically adjusts the value of the two others. The α-parameter can take values between 0 and $-\frac{1}{2}$, for which $\alpha = 0$ give zero numerical damping, while the highest level of numerical damping is introduced for $\alpha = -\frac{1}{3}$ [8].

In Abaqus, the time increments are slightly changed in-between the solution steps according to convergence criteria. In general, this introduces some numerical noise in the solution for high frequencies. An advantage of introducing some numerical damping, i.e. setting the α-parameter different from zero, is that this noise is efficiently removed [7].

As earlier discussed, evaluating the time increment size and stability of the operator is important with respect to convergence. However, this is not a major issue for the implicit schemes in Abaqus, all of which are unconditionally stable [8]. Moreover, the default setting for nonlinear procedures in Abaqus is that time increments are set and adjusted automatically during an analysis, for implicit schemes through satisfying the convergence requirements for the Newton solver iterations. Slow convergence is, for instance, counteracted by reduction of the time increment. [8].

3.4 Finding eigenfrequencies and measuring damping

The eigenfrequencies and the damping ratio are basic parameters to measure when assessing how damping applied to a model is working. The preliminary work mentioned in section 1.3 tested strategies for finding eigenfrequencies and measuring damping in Abaqus. The choice was to use the complex frequency step, which deliver both the eigenfrequencies and the corresponding damping ratios. The eigenfrequencies were checked towards analytical results from the eigenvalue problem and the damping ratios were checked by running a time integration and using the logarithmic decrement method. Both the eigenfrequencies and the damping ratios showed to be correct. On this basis, the complex frequency step was chosen as the tool for finding eigenfrequencies and measuring damping ratios in this work.

3.4.1 Eigenfrequencies

The eigenvalue extraction in Abaqus is done by a frequency step. This is based on solving the eigenvalue problem, given in Equation (3-7) [8].

$$(\mathbf{K} - \omega^2 \mathbf{M})\boldsymbol{\phi} = 0 \qquad (3\text{-}7)$$

Here, **K** is the stiffness matrix, ω the eigenfrequency, **M** the mass matrix and $\boldsymbol{\phi}$ the eigenvectors. The solving of this eigenvalue problem is, for comparison, also done by hand calculations, which are performed in MATLAB (an example script is found in Appendix A). Equation (3-8) is used to calculate the analytical eigenfrequencies, which also is done in MATLAB. The expression in Equation (3-8) yields when solving the beam equation based on Euler-Bernoulli beam theory [6].

$$\omega_n = n^2\pi^2\sqrt{\frac{EI}{\rho A l^4}} \tag{3-8}$$

Here, ω_n is the eigenfrequency number n, n the number of the mode, E the elastic modulus, I the second moment of inertia, ρ the density per unit length, A the cross-sectional area and l the span length.

3.4.2 Measuring damping

Rayleigh damping was applied at the material level in the Abaqus model. As discussed in section 2.2.1 this is a proportional damping model based on the following equation:

$$\mathbf{C} = \alpha_{\mathrm{R}}\mathbf{M} + \beta_R\mathbf{K} \tag{3-9}$$

Here, **C** is the stiffness matrix, $\alpha_R$and $\beta_R$the Rayleigh-coefficients, **M** the mass matrix and **K** the stiffness matrix. To compare the output from Abaqus, hand calculations were also done for the damping in MATLAB (Appendix A). Then the same Rayleigh-coefficients were applied as in Abaqus and the damping matrix was diagonalized by eigenvectors to find modal damping.

In Abaqus, the damping was outputted by two methods: using the logarithmic decrement method on a free vibration plot from a time integration analysis, and results from the complex frequency step. These two methods were also compared to each other. The logarithmic decrement method is a commonly applied method for measurement of damping and is explained in section 2.3.

The complex frequency step solves the complex eigenvalue problem and output complex eigenfrequencies and a parameter denoted “effective damping ratio”. This method is very easy to use and is very fast, compared to the logarithmic decrement method, which require a time integration. The complex frequency step must be combined with a preliminary frequency step prior to the analysis in Abaqus. This is due to the extracted modes in the frequency step are applied through the complex frequency step. This leads the number of modes extracted in the frequency step to be the upper boundary for the number of complex eigenfrequencies [8].

An important output parameter from the complex frequency step is the formerly mentioned effective damping ratio. In the Abaqus documentation, it is defined as: [8]

$$Effective\ damping\ ratio = -2\frac{\alpha}{|\omega|} \tag{3-10}$$

For this parameter, α is the real part of the eigenvalue and ω the imaginary part of the eigenvalue. These complex eigenvalues can be written as $\lambda_n = -\zeta_n\omega_n \pm i\omega_{nD}$ [4]. Here, ζ_n is the damping ratio, $\omega_n$the eigenfrequency and ω_{nD} the damped eigenfrequency. For the situation at hand, with small damping levels, it is common to assume $\omega_{nD} \approx \omega_n$, due to the relation $\omega_{nD} = \omega_n\sqrt{1-\zeta_n^2}$ and ζ_n being small. By applying this approximation to Equation (3-10) one can see that the effective damping ratio may also be written as only $2\zeta_n$. This damping ratio, ζ_n, is commonly used to describe the damping of a structure. It is given as $\zeta = \frac{c}{2m\omega}$, where *c* is the damping, *m* the mass and ω the eigenfrequency. Summing this up, it is shown that for small amounts of damping, the half of the effective damping ratio is approximately equal to the damping ratio, ζ.

The three methods for measuring damping were compared. These were: hand calculations of Equation (3-9), the logarithmic decrement, and the complex frequency step. This showed logarithmic decrement and hand calculations to be almost equal and the effective damping ratio to be approximately two times any of the latter, which is correct according to what was shown in the previous paragraph. On that basis the complex frequency step has shown accurate results and is clearly the easiest and fastest method. Therefore, it was chosen as the method for measuring damping.

4 Beam model

As a starting point, a simple simulation was sought to test how different choices affect damping in a model. The simply supported beam has a simple analytical solution, is easily modelled and calculated by software and is a common component in structures. By this, the simply supported beam was chosen as a starting point. A beam can further be modelled with a basis in different theories: beam, shell or solid volume theory. All these three can be reasonable choices under different circumstances. Therefore, all of them were examined.

In modelling procedures, it is important to assure stability when it comes to the choice of element type and mesh size. Different elements where analysed and results compared to hand calculations. Sufficient convergence for element meshes were confirmed and determined the choice of mesh size.

A previous study by Labonnote [2] indicate that shear deformations influence damping in some way. Additionally, the shear stiffness of timber is relatively much lower than the bending stiffness. Due to this, models that consider shear stiffness and shear deformations was wanted. Focusing on this, the analysis was limited to four different elements. Timoshenko beam theory, Mindlin-Reissner theory and solid volume theory is the basis for three of the elements chosen and fulfil the requirements regarding shear [7]. The fourth element is a beam element based on Euler-Bernoulli theory, which does not take shear into account. By including this element, the influence of shear in the Timoshenko element can possibly be assessed in a better way.

4.1 Basis for damping implementation

Previously, the authors have done a preliminary work on damping in Abaqus. Different strategies to implement damping and to assess the results were tested. The conclusion was to test out the Rayleigh model more extensively and try to combine it with other models. This was due to the flexibility of being able to implement Rayleigh damping as a material feature, such that different levels of damping can be introduced to different parts.

Even though a viscous and frequency dependent damping model like Rayleigh cannot give a comprehensive description of damping behaviour, it could be sufficient. Further, the Rayleigh model showed stable and consistent results when run with different damping levels, different element types and meshes. From experimental measurements of similar beams to what is being tested here, it has been found that the first and second vertical vibration mode is damped about 1 % and 1.5 %, respectively [2]. From this, two Rayleigh-coefficients were calculated (example calculations are found in Appendix A) and these has been held constant in all the work with the beam models here.

4.2 Creating the models in Abaqus

The models used in this thesis where created and analysed in the finite element software program Abaqus/CAE (from now on only referred to as “Abaqus”). How these models where created is presented here. The different steps are ordered as suggested by following the model tree (the list at the left side of the user interface) in Abaqus from top to bottom. A complete guide for the 3D beam model with Euler-Bernoulli elements is provided. For the other models used in the thesis only partial guides are made. These partial guides are covering the deviating sections of the modelling procedure.

4.2.1 Euler-Bernoulli beam element models – B33

At the start of the session a “Standard/Explicit model” is chosen.

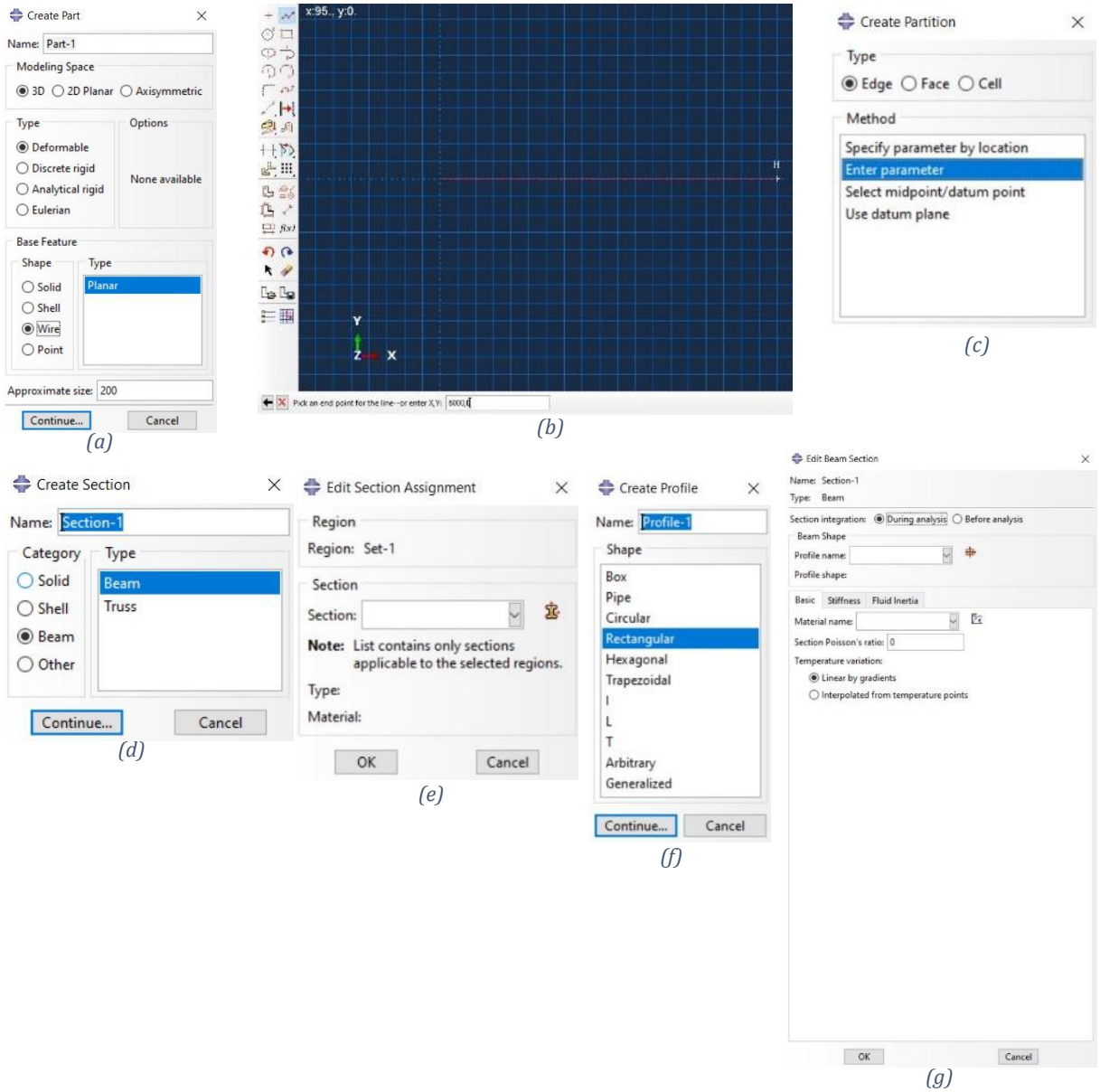

Figure 4.1a-g: Start of modelling with creation of part, partition, section and profile. The modelling procedure follows the letter numbering presented.

The next step is to create a part, this and the following choices are shown in Figure 4.1. Then a choice must be made on what type of model is wanted, concerning dimensionality, type and base feature. For this case, with a three-dimensional Euler-Bernoulli beam, “Modelling space” is set to *3D*, “Type” is set to *Deformable* and “Base feature” is set to *Wire* and *Planar*. From here a space for sketching the model is entered automatically. For a beam model, a line representing the centreline of the beam must be sketched. This can be done by the tool “Create lines: Connected”, which allows for picking points with the cursor or entering start and end points by values. To prepare for a prescribed deformation of the midpoint the beam is partitioned in two. The partition is created by “Type” as edge and “Method” as enter parameter. By setting the parameter to *0.5*, the beam is partitioned in the middle. Then a set can be created from the point in the middle of the beam to easily refer to this point.

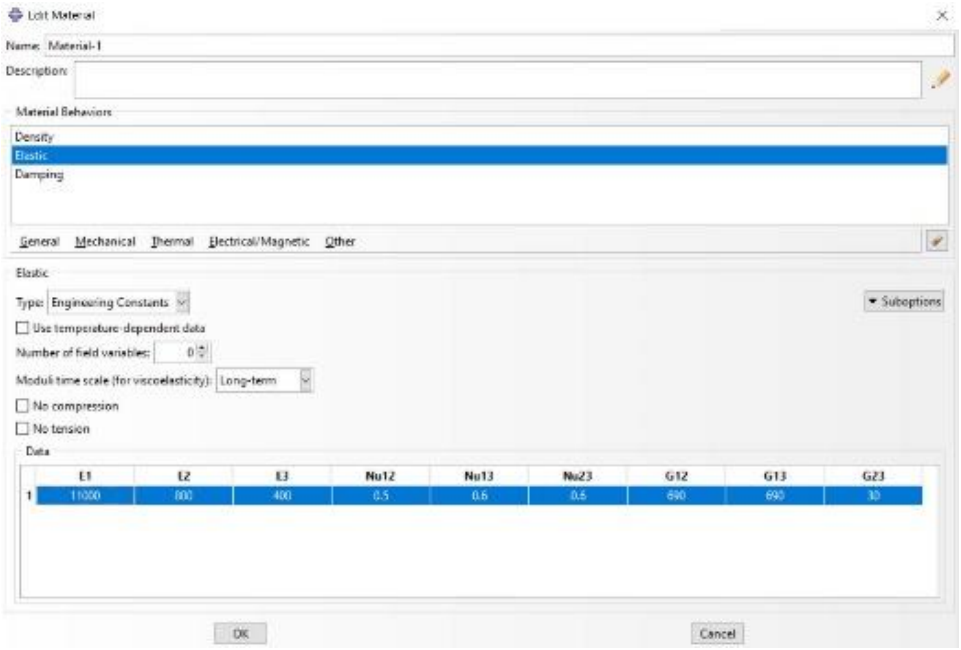

Figure 4.2: The three material behaviours density, elastic and damping are configured.

Further, a section is to be assigned. To assign a section, a profile and a material is required. For the section, the option beam is set for both “Category” and “Type”. The profile is made with “Shape” *rectangular*. Three options are defined for “Material behaviours”: *Elastic*, *Density* & *Damping,* this is shown in Figure 4.2. For the elastic material behaviour “Type” is set to *Engineering Constants*, which yields the opportunity to define an orthotropic material by elastic modules, shear modules and Poisson ratios for all dimensions. The chosen data for the material is displayed in Table 4.1, and is chosen to simulate the common C24 quality of Norway Spruce [14], [15].

Table 4.1: Data for defining the material used in the models of the beam.

E1 [N/mm²]	E2 [N/mm²]	E3 [N/mm²]	G12 [N/mm²]	G13 [N/mm²]	G23 [N/mm²]
11 000	800	400	690	690	30
Nu12 [ ]	Nu13 [ ]	Nu23 [ ]	Density [kg/m³]	Alpha, α_R [ ]	Beta, β_R [ ]
0.5	0.6	0.6	420	1.13	7.47×10^{-5}

Defining a beam orientation for the whole beam is next. This is done by the tool “Assign beam orientation” found in the module “Property”. The default vector “0.0,0.0,-1.0” is accepted.

Under “Assembly” an instance is created with “Instance Type” set to *dependent*.

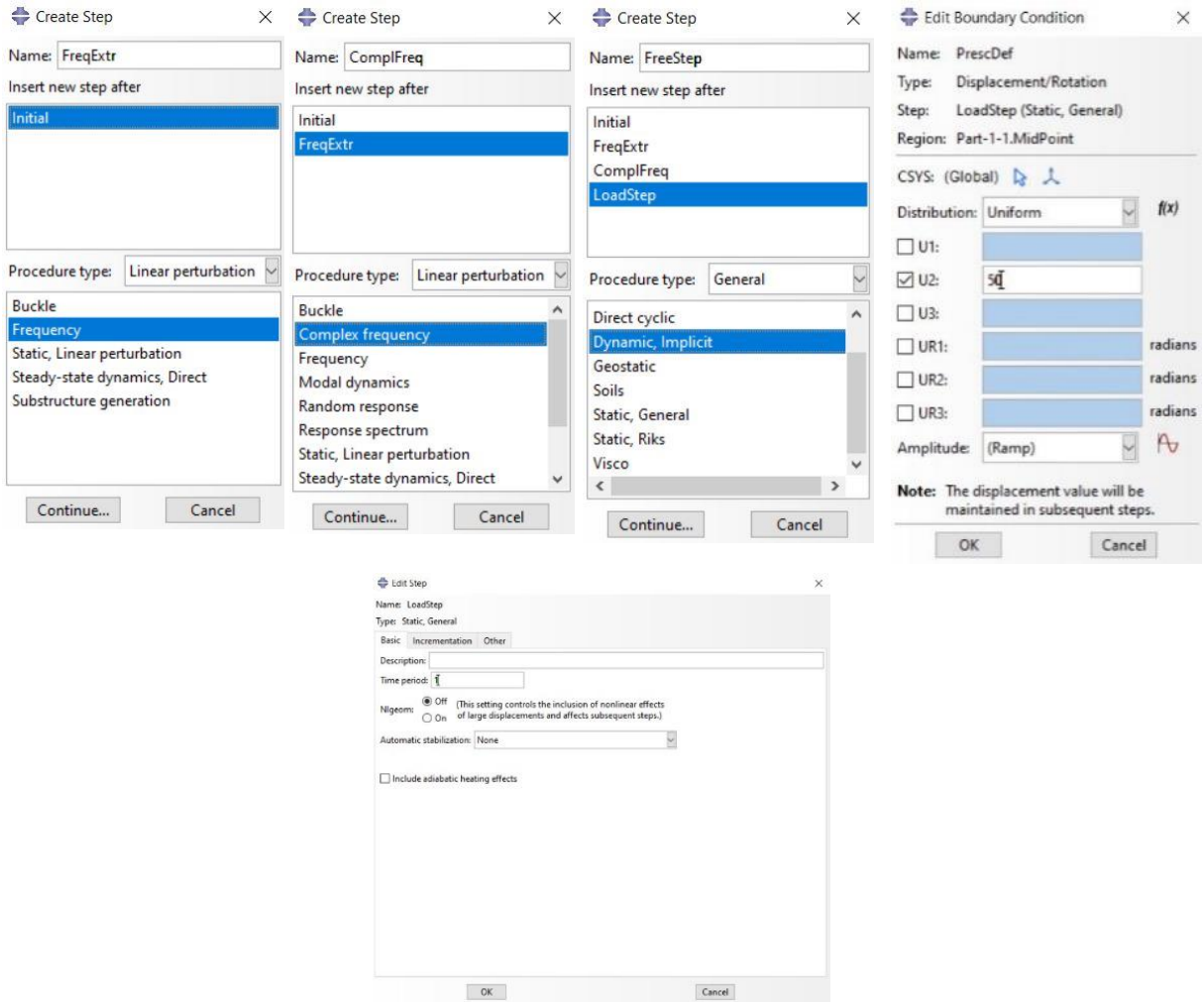

Figure 4.3: The steps created for the analysis and the prescribed deformation.

What will be analysed is determined next; this is done by creating different steps. The steps created are displayed in Figure 4.3. For extraction of complex eigenfrequencies and effective damping ratios, the complex frequency step is required, and prior to this, a frequency extraction must be done by the frequency step. In the frequency step “Maximum frequency of interest” is set to *100*. As the modelling is done in seconds this

corresponds to 100 Hz. The rest is set to default, as is also the whole complex frequency step. For the time integration the general step *Dynamic, Implicit* is applied. To set a prescribed deformation a *Static, General step* is defined. In this step a boundary condition, of "Category" *mechanical* and "Type" *displacement/rotation*, is created. In the editing, the set for the midpoint is chosen as region and "U2" is given the value 50.

After defining the prescribed deformation, the *Dynamic, Implicit* step is created. In the "Basic" tab the time period is set to 2. "Incrementation" tab is edited so that "Maximum number of increments" is *4 000*, "Initial increment size" is *0.001* and "Minimum increment size" is *0.0005*. These choices are made for the model to converge. Dependent upon the model, greater or smaller increment sizes should be applied. Additionally, to have a decent number of points between each amplitude peak in the response curve, the increments cannot be too large. This is dependent upon what is the frequency for the first vertical mode. Lastly, to remove numerical damping, in the "Other" tab, "Time integrator parameter" is specified to *0*. To have the beam vibrating freely the boundary condition with the prescribed deformation must be deactivated in this step. This can be done by right clicking it in the model tree under the free step and choose "Deactivate".

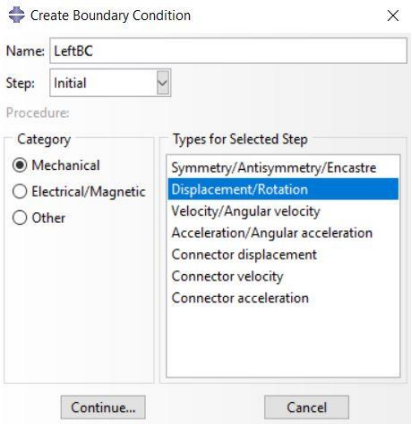

Figure 4.4: Configuration of the beam's boundary conditions.

The beam is modelled as simply supported. For this, a boundary condition on each end of the beam is required. These are of "Type" *displacement/rotation,* and are created in the step "Initial", as in Figure 4.4. At one end *U1*, *U2* and *U3* is locked, and at the other end, *U2* and *U3* is locked.

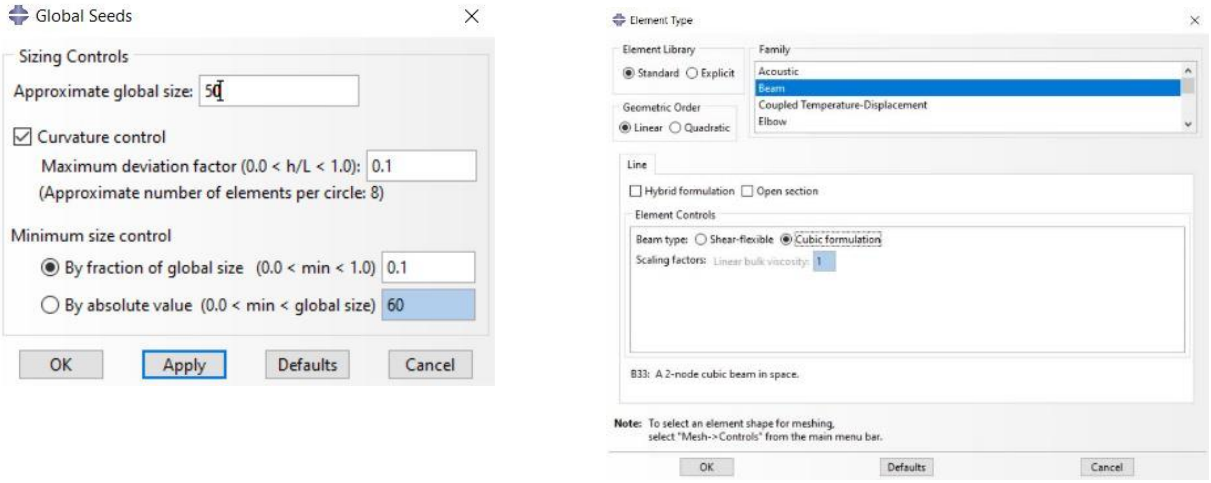

Figure 4.5: The choices made when meshing the beam.

From the tool "Seed part" the beam is meshed by global seeds. "Approximate global size" is set to *50*. The element type is specified by the tool "Assign element type", in which *cubic formulation* is selected for "beam type ". These choices are shown in Figure 4.5.

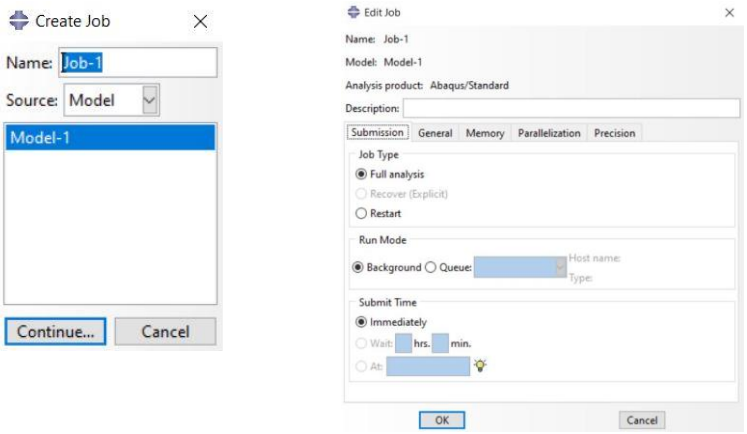

Figure 4.6: The last part of the analysis is to create a job and run it.

In the end a job is created and submitted, for which all default choices can be accepted. Figure 4.7 shows the dialog boxes for creating and editing a job.

4.2.2 Timoshenko beam element models – B31

To create a beam model based on Timoshenko beam theory there is only one step to be changed from the original beam model. In the tool "Assign element type" the option *Shear-flexible* must be chosen.

4.2.3 Conventional shell element models – S8R

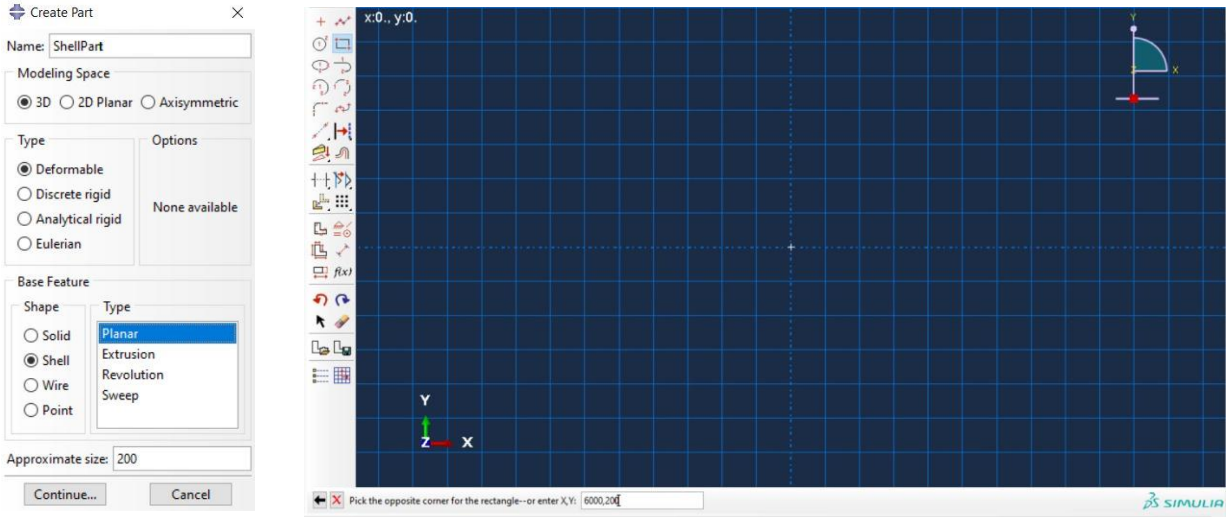

Figure 4.7: Creating and sketching the part for the shell model.

Figure 4.7 shows creation and sketching for a part. For a shell model the part must be created differently from the beam model. The differing choice is to set the "Base Feature" and "Shape" to *Shell* and further, "Type" to *Planar*. When sketching the shell, one must decide upon whether the thickness of the shell shall represent the beam height or the width. The dimension that is not represented by the thickness is drawn in the sketching area. Here, both the alternatives are tested in different models.

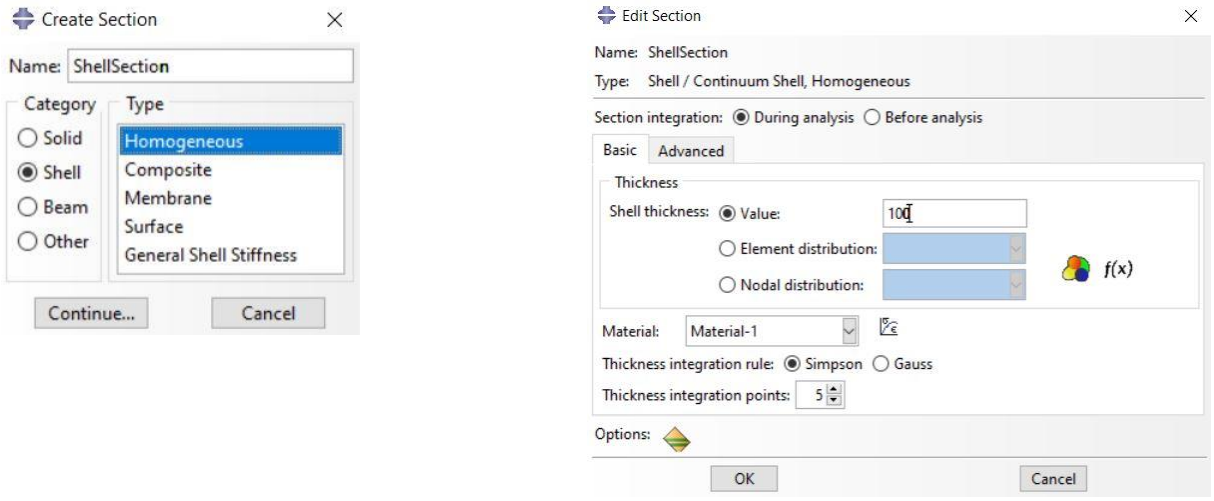

Figure 4.8: Creating a section for the shell model.

For the section "Category" is set to *Shell* and "Type" is set to *Homogenous*. In the dialog box for editing the section, the shell thickness is set based on what dimension it shall represent (height or width). This is shown in Figure 4.8.

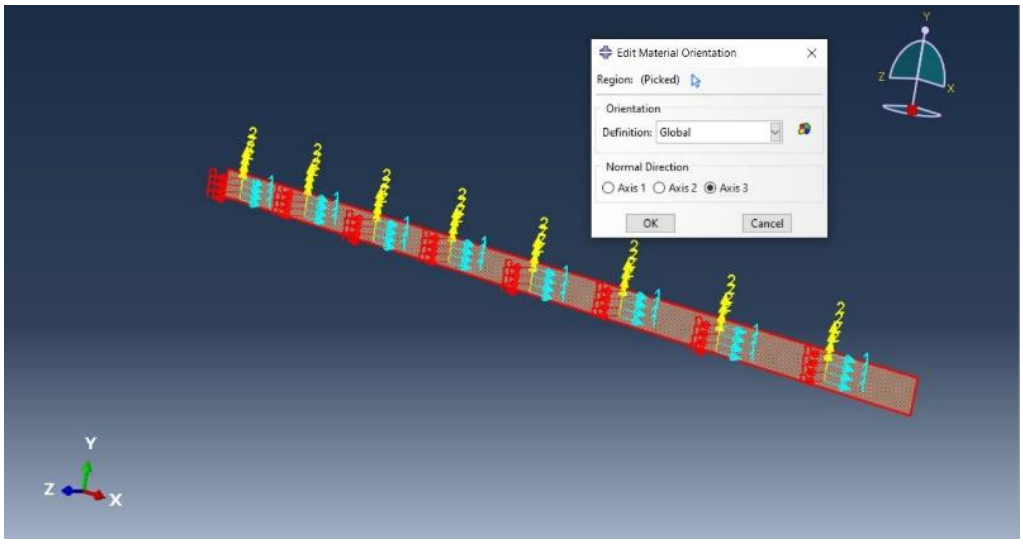

Figure 4.9:The material orientation previewed on the model.

Instead of applying a beam orientation, a material orientation is specified, as in Figure 4.9. This is done for the whole part with the 1-axis as longitudinal and the 2-axis as the second axis in the sketched plane.

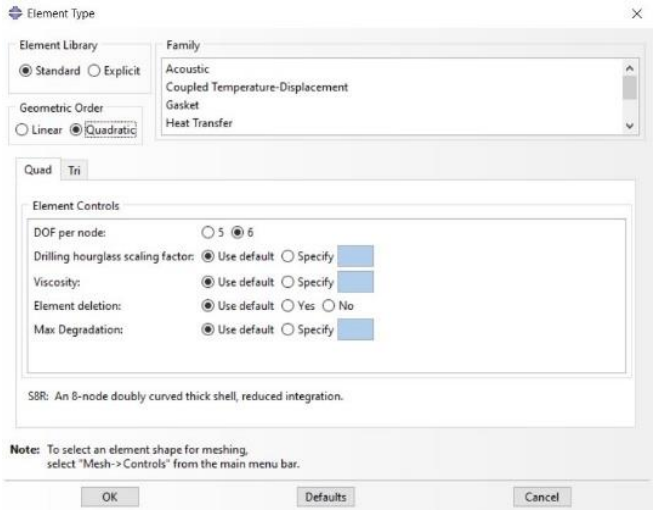

Figure 4.10: Choosing the preferred element type.

During meshing, the tool “Assign element type” is used to change element type. The dialog that opens is shown in Figure 4.10. “Geometric order” is set to *Quadratic*, and this yields a Mindlin-Reissner based element (“thick shell theory”) with reduced integration, named S8R in Abaqus.

Remark that for the shell model, boundary conditions should be adjusted according to shell orientation (shell thickness used to represent beam height or width) and whether vertical or horizontal vibration modes are analysed.

4.2.4 3D solid element models – C3D20R

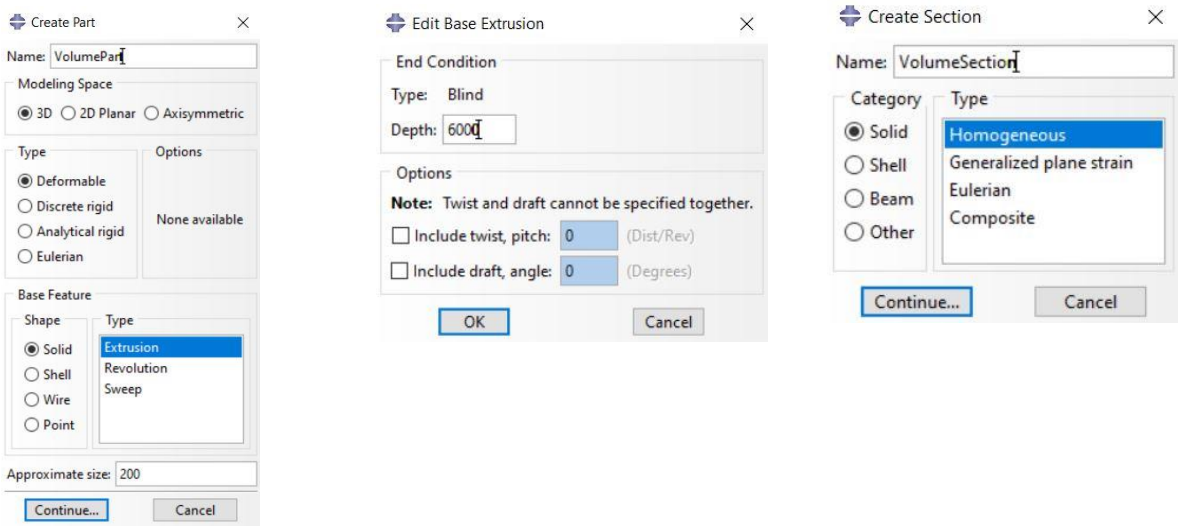

Figure 4.11: The initial choices for the 3D solid model.

For the 3D solid model, there are deviations from the beam model when creating the part. The choices made for this model is presented in Figure 4.11. Under “Base feature” the “Shape” is set to *Solid* and “Type” is set to *Extrusion*. After this, a sketching of the beam’s section is made, and when the sketch is accepted the prompt for extrusion depth is set equal to the beam length. In the creation of the section, the “Category” is set as *Solid* and “Type” as *Homogeneous*.

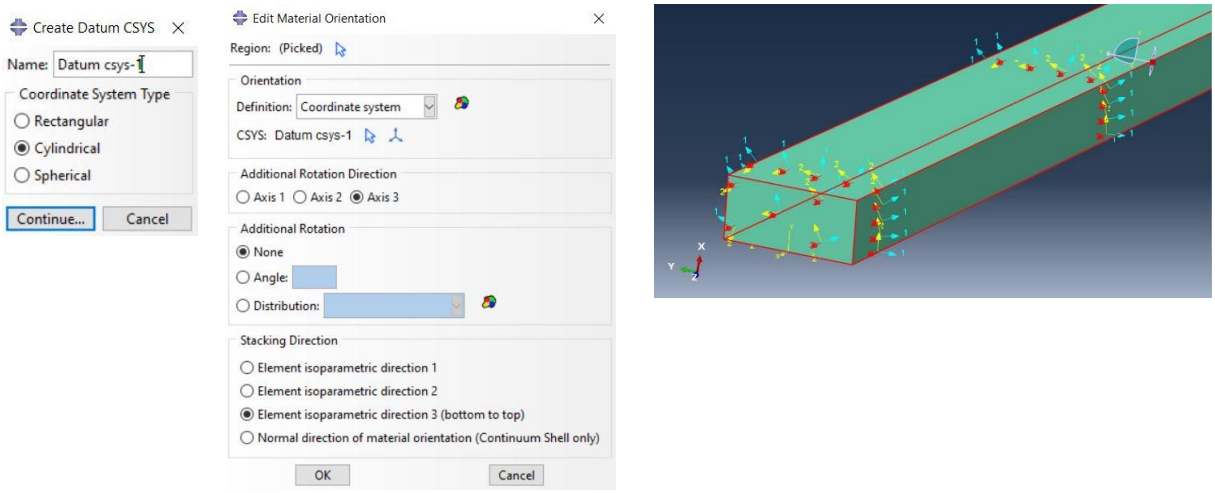

Figure 4.13: Material orientation is defined by a user-created coordinate system.

Instead of a beam orientation, a material orientation is specified for the 3D solid model, as in Figure 4.13. If the beam is made from solid timber the mathematically most representative model is a cylindrical material model. This is made by defining a local coordinate system, which can be done by the tool “Create Datum CSYS: 3 points” found in the module “Property”. The material orientation is then configured with “Definition” set to *Coordinate system* and “CSYS” as the user-created coordinate system. This can be configured by selecting “Datum CSYS List…” in the prompt area, this prompt shows up after selecting the beam as the target for defining a material orientation.

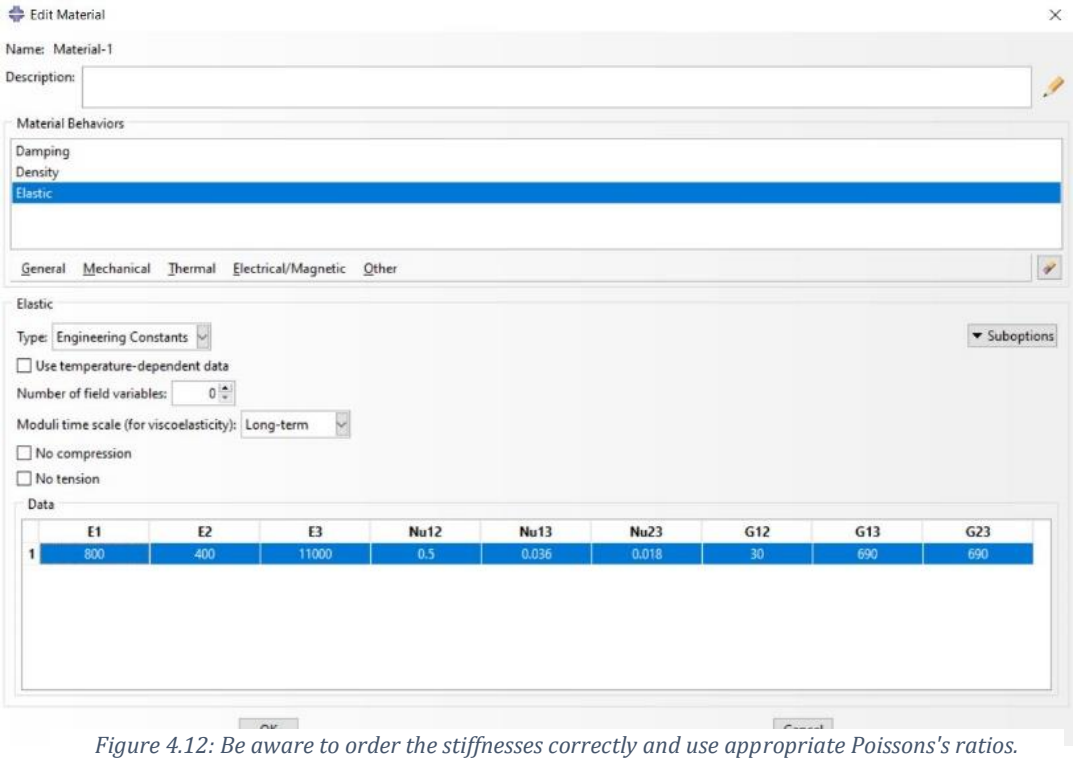

Figure 4.12: Be aware to order the stiffnesses correctly and use appropriate Poissons's ratios.

The material definition must be edited when the cylindrical material orientation is configured as described in the above section. Now the longitudinal axis of the beam is axis

"3", axis "1" is radial and axis "2" is tangential. It is then important to rearrange the values for the stiffnesses accordingly and calculate new Poisson's ratios. The rearrangement is done in the module shown in Figure 4.12. The Poisson's ratios are calculated by $\nu_{ij} = \nu_{ji}\frac{E_{ii}}{E_{jj}}$ [16].

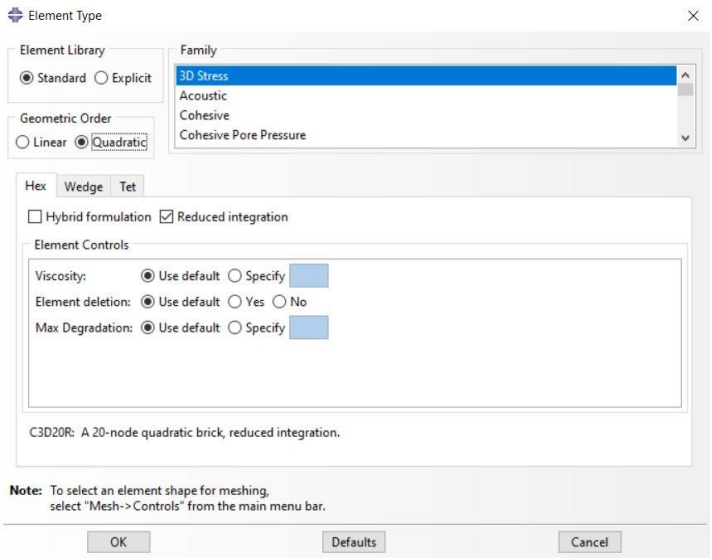

Figure 4.14: The element type is chosen when the model is meshed.

During meshing, the tool "Assign element type" is used to change element type, as shown in Figure 4.14. "Geometric order" is set to *Quadratic* and in the tab "Hex" the box referring to reduced integration is checked. This yields a 20-noded element with reduced integration, named C3D20R in Abaqus.

For the 3D solid models, there are additional boundary conditions due to the 3D extent of the elements. The edges of the end faces are supported by fork-supports.

4.3 Examining with iSight

A process of exploring how changes in different parameters influence a single output or multiple outputs can be time-consuming. To alleviate this, the software iSight was used. This software can automate processes and handle input and output between different software components or just a single component. One of the supported components is Abaqus. iSight was used to automatically run a range of analyses in Abaqus without needing to manually input and run every analysis.

iSight require a model to be made in Abaqus as a starting point to run simulations. The models used from Abaqus are created as described in the previous sections. There was set an upper limit on eigenfrequencies at 100 Hz since frequencies above this is not relevant for the structural analyses.

4.3.1 Element meshes in iSight

When different beam lengths and beam heights are run from the same Abaqus model in iSight, a choice must be made on how the different beams are to be meshed. For the two beam models and the shell models, the option with global seeds was chosen. Then a size for the elements in the mesh are chosen, so that longer beams will have more elements than shorter beams. In the 3D solid model, the number of elements can vary a lot as the beam and the elements has an extent in space. This was avoided by choosing to seed the edges of the beam. This was done to the length, height and width. In this way the element mesh was kept at a constant 5x10x80, five in the width, 10 in the height and 80 in the length.

4.3.2 Modifying boundary conditions to simplify output review

To simplify the comparison of the same modes for different analyses with different beam-configurations and element types, it was considered more practical to investigate modes when separated in types of modes, i.e. vertical, horizontal and torsional. However, this involves changing the boundary conditions between investigations of different modes in their respective dimension. Here, horizontal modes were isolated by fixing the vertical degree of freedom (DOF), referred to as U2 in Abaqus, along the beam length. The equivalent was done for vertical modes by fixing the horizontal out-of-plane-DOF, U3. For both cases contributions of torsion in the eigenmodes were excluded by locking rotational DOFs, UR1, about the beam axis.

Sample tests were carried out, both with and without the respective DOFs locked along the beam length, to see if the isolation of the modes affected the resulting eigenfrequencies or damping ratios. They were performed for both the beam models, the shell models and the 3D solid models. The validation-models displayed either small or no differences in results for eigenfrequencies and damping ratios. The examined modes that were purely horizontal or purely vertical was not changed when isolated. The modes that initially seemed to have contributions from torsion did display changes when torsion was restricted, although, these changes were minor.

The relative changes between test samples were small (about 1%), both with respect to eigenfrequencies and effective damping ratios. This indicated that introducing respectively horizontal and vertical constraints in order to investigate vertical and horizontal modes separately, was not unreasonable.

Additionally, rotation about the longitudinal axis was restrained. This excluded all torsional behaviour from the eigenmodes. This had no effect on the 3D solid model as there are no rotational DOFs, but it affected the other element types. The effect of this was examined by sample testing and the difference in corresponding eigenmodes (on both frequency and shape) was small. The difference was e.g. 0.5-1Hz for some shell element configurations.

With respect to torsional modes, it may be argued that a state of torsional restriction for beams in structural assemblies to some degree is more realistic. The behaviour of a beam in a structural assembly is typically related to bending modes of the beam, not torsional. Excluding torsional modes from investigation therefore does not violate the intents here. Rather, the results should be more realistic. Additionally, torsional modes typically appear for higher frequencies – typically higher than the first two horizontal and vertical modes – and is therefore considered as less relevant with respect to structural properties.

4.3.3 Setup of the model in iSight

The model in iSight were set up such that small changes were made for each analysis run in Abaqus. The model thus goes through a range of beam heights that are analysed for different beam lengths. The output that is reviewed is the eigenfrequencies and the damping ratios reported by the complex frequency step in Abaqus. In the following the creation of the basic model in iSight is elaborated.

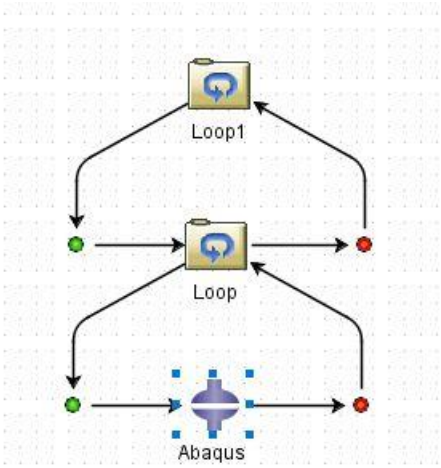

Figure 4.15: Overview showing loop components and a Abaqus component, found in the "Sim-flow" tab of iSight.

The basic model was made with two loop components, found in the tab “Process Components”, and one Abaqus component, found in the tab “Application Components”. A visual representation of this is given in Figure 4.15. The Abaqus component gives access to different input values, in this case the beam length and the cross-sectional beam height, also referred to as the beam height. For the loops, the bottom loop is inside the other so that all the iterations of the bottom loop will be run for each iteration of the top loop. In this case the top loop will change the beam length and the bottom loop will change the beam height.

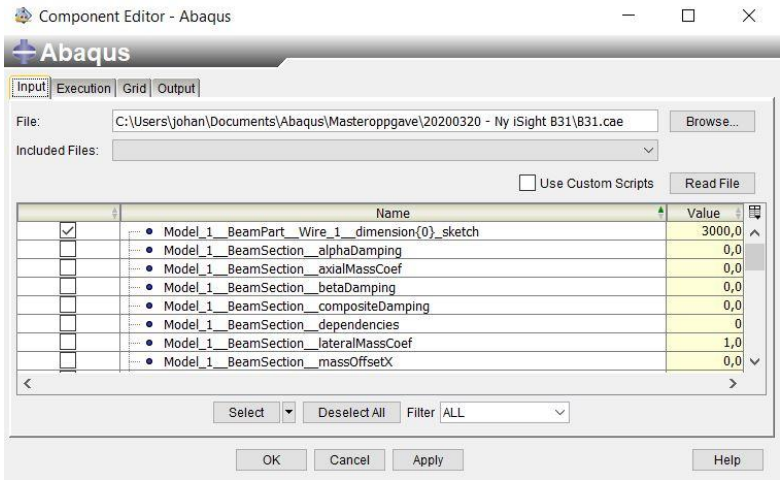

Figure 4.16: Configuring the Abaqus component in iSight.

Starting off, a model was created and analysed in Abaqus. The .cae-file was then read in the input tab of the component editor in iSight, as shown in Figure 4.16. Then the boxes referring to the beam length and the beam height was checked. In the output tab the .odb-file associated with the job initially analysed in Abaqus was read. The two boxes referring to matrices for damping ratios and eigenfrequencies were checked.

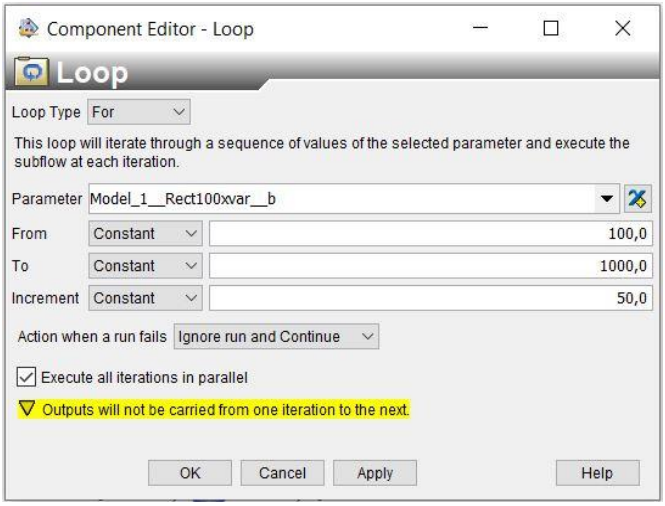

Figure 4.17: Configuring the bottom loop for the beam height.

The bottom loop was set to iterate over different beam heights. This was configured in the component editor by choosing the relevant parameter from the list in “Parameter”. Then “From”, “To”, “and “Increment” was set to 50, 1000 and 50, respectively, as shown in Figure 4.17.

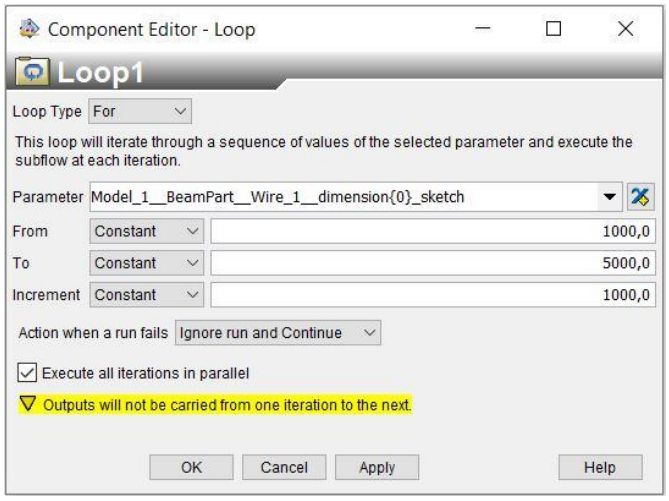

Figure 4.18: Configuring the top loop for the beam length.

The configuration of the top loop is shown in Figure 4.18. For this loop the parameter was set so that it iterated over the beam length. "From" was set as 1000, "To" as 5000 and "Increment" as 1000. By these choices, the beam lengths 1000, 2000, 3000, 4000 and 5000 (Abaqus is unitless, but lengths correspond to mm here) were analysed.

4.3.4 How the output is reviewed

In all the models, the same level of Rayleigh damping was applied. This makes all the different geometrical configurations and element type configurations of the beam comparable to the same mathematical representation of Rayleigh damping. This representation is based on Equation(2-12), repeated here for convenience [3]:

$$\zeta_i = \frac{\alpha_R}{2\omega_i} + \frac{\beta_R \omega_i}{2} \tag{4-1}$$

The damping ratio, ζ_i, can be expressed as a function for set values of both α_R and β_R with the circular frequency, ω_i, as the argument. By this, the damping ratio in Equation (4-1) is plotted as a curve, referred to as the "Rayleigh-curve". Results from the complex frequency analysis is presented in the same plot. These results are presented as single data points, where one point represent the modal damping ratio or the eigenfrequency for a mode of a specific beam configuration. This comparison is made since Rayleigh damping per definition, as given in section 2.2.1, is a modal damping type. As a result, the Rayleigh-curve express damping levels per mode for the different frequencies.

To check for possible discrepancies in the results, the eigenfrequencies from the Euler-Bernoulli beam theory-based equation, referred to as the analytical eigenfrequencies, were plotted along with the eigenfrequencies from the complex frequency analysis in Abaqus. These analytical eigenfrequencies are explained in theory section 2.1.4. These eigenfrequencies are plotted both undamped and damped to have greater comparability to the results from Abaqus.

4.3.5 Horizontal vibration modes

The iSight-simulations where executed with iteratively increasing beam height and beam length. From the analytical expression for eigenfrequencies based on beam theory given in section 2.1.4, one can find the following relation for the beam: with the same density, stiffness and length, the eigenfrequency is only affected by changes in beam height, i.e. the width is not relevant. Regarding horizontal modes, one can then see that eigenfrequencies should be the same when only the beam height is increased.

The results confirmed this by showing very small, or no difference, in frequencies and damping ratios between the horizontal modes for the beams of the same length. Therefore, these results are not so interesting in comparison to the vertical vibration modes and are not presented here. For completeness, these results are included in Appendix E.

4.3.6 Euler-Bernoulli beam element models – B33

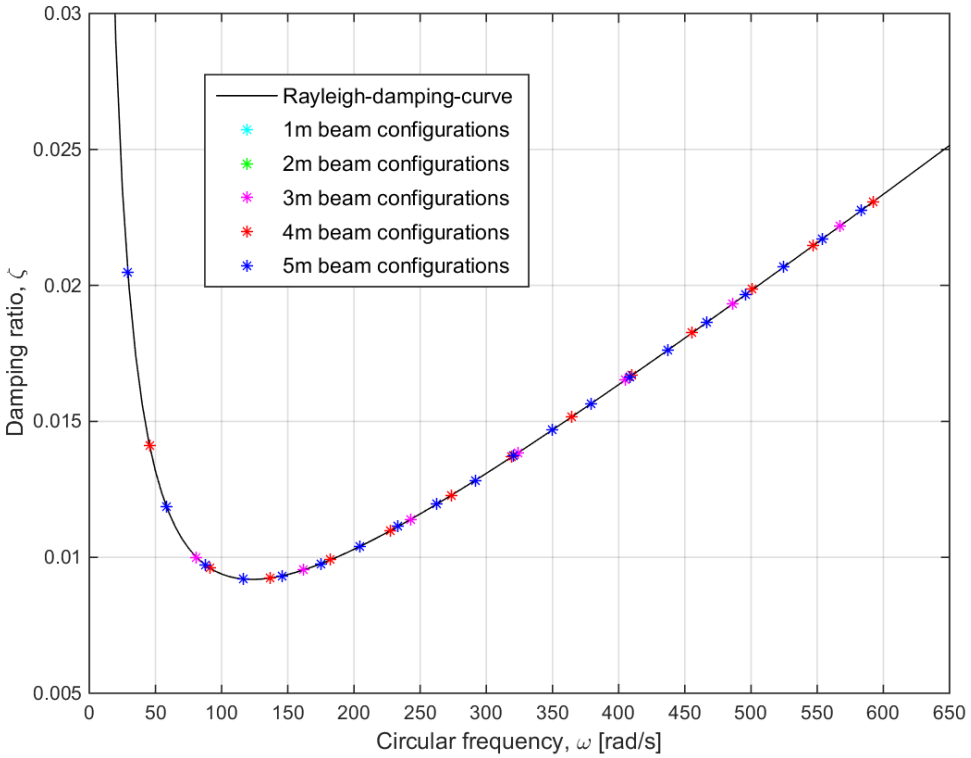

Figure 4.19: Euler-Bernoulli beam model. Damping from Abaqus plotted towards the Rayleigh-curve. First vertical mode of the beam.

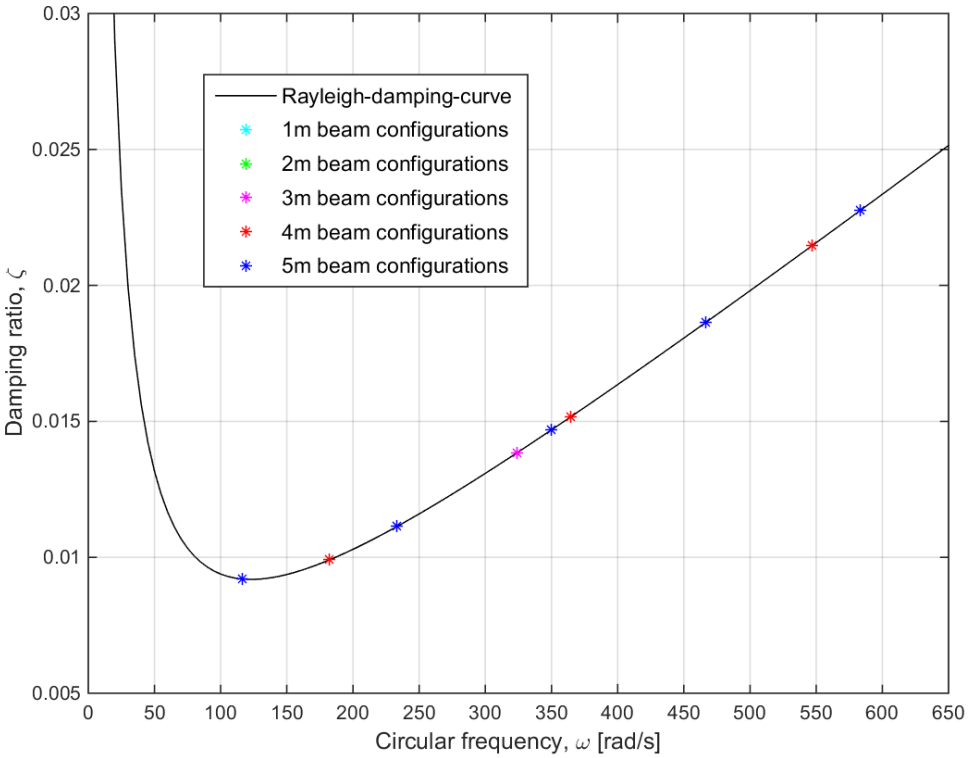

Figure 4.20: Euler-Bernoulli beam model. Damping from Abaqus plotted towards the Rayleigh-curve. Second vertical mode of the beam.

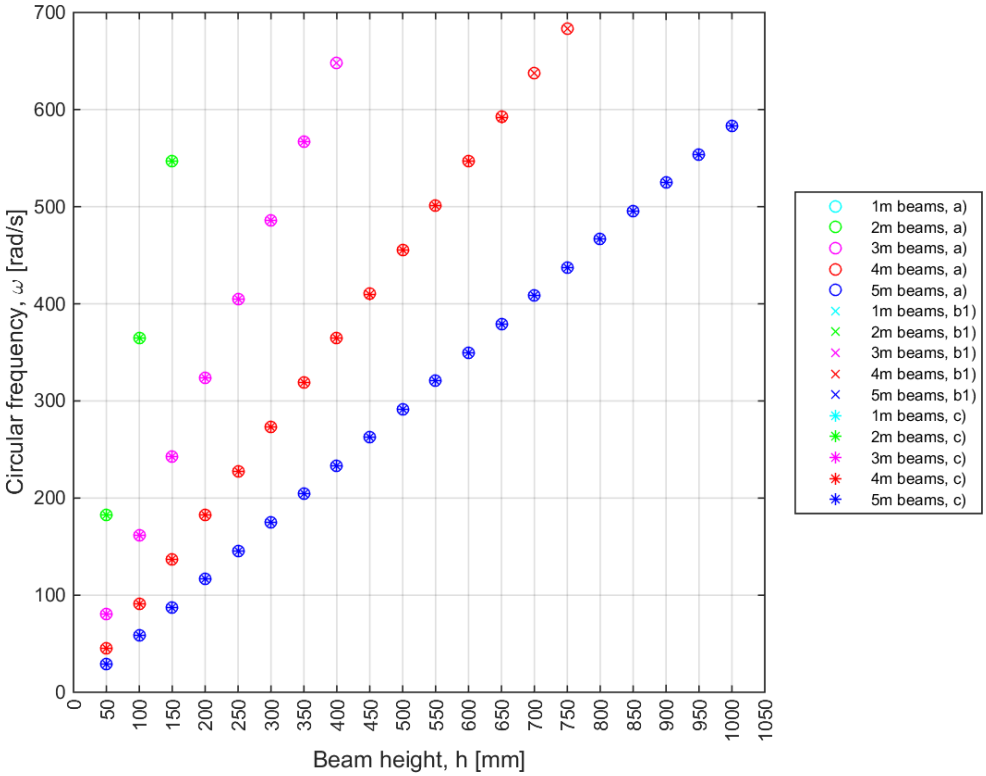

Figure 4.21: Euler-Bernoulli beam model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Analytical damped by 2% and c) Complex frequencies from Abaqus.

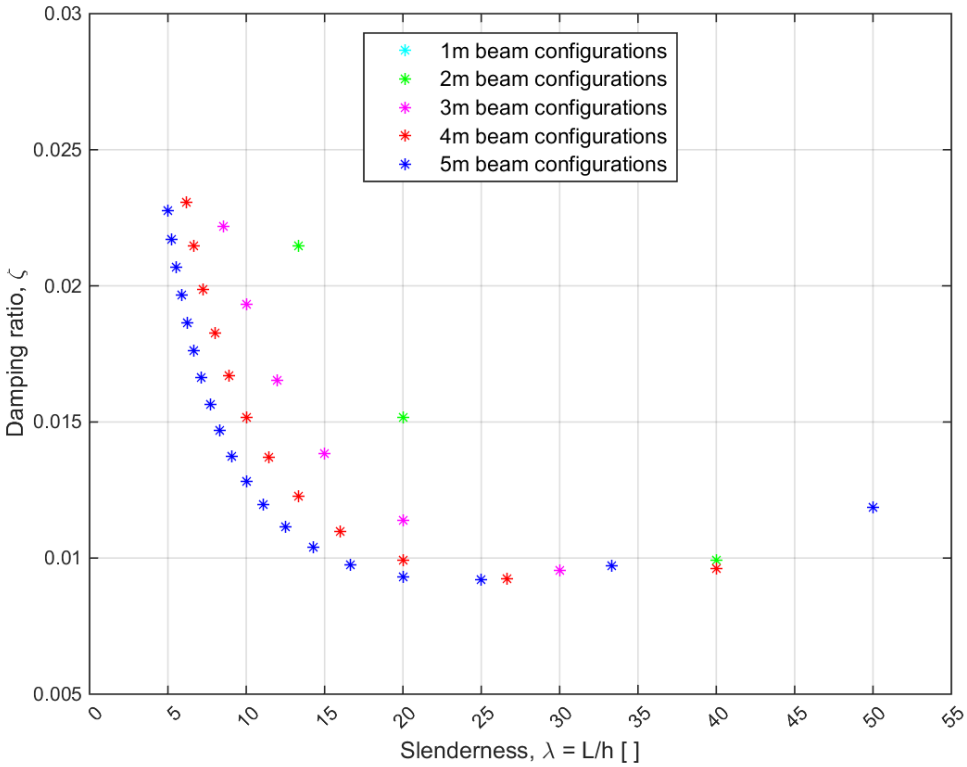

Figure 4.22: Euler-Bernoulli beam model. Damping from Abaqus plotted for the slenderness. L = span length, h = beam height.

In Figure 4.19 and Figure 4.20, the results for the first and the second vertical mode, respectively, are presented. The damping ratios from Abaqus are plotted as different coloured stars. One single data point (star) represents one beam with a specific beam length and beam height. The damping ratio is plotted for the eigenfrequency of the mode, the frequency being given in radians per second (rad/s). The plot-values are sorted by colour according to beam length. This is the case for all plots of results for the beam models that is presented in the rest of this text. As seen in the plots from the Euler-Bernoulli beam, there are no values for the beams that are one meter long. This is due to an upper limit of 100 Hz that was set for maximum eigenfrequencies extracted. The one-meter long beams are included in the analysis, but they are too stiff for the 100 Hz limit, thus, results are not obtained for these cases.

One can see that all the damping ratios match the Rayleigh-curve (described in section 4.3.4) by following it for all the resulting eigenfrequencies. The upper limit for the maximum frequency set at 100 Hz leads to few results for the second vertical mode.

As seen from Figure 4.21 the analytical undamped eigenfrequencies, analytical frequencies damped by 2% and the resulting damped frequencies from Abaqus all increases linearly for increasing beam heights. This was expected, since the damped frequencies are approximately equal to undamped frequencies for low levels of damping, and the undamped frequencies varies linearly with the beam height, h.

The damping ratios are also plotted for the dimensionless beam slenderness, $\lambda = \frac{L}{h}$, where L is the beam length and h is the beam height. This is displayed in Figure 4.22. In this plot the different beam lengths follow the same trend regarding damping ratio but are not equal, i.e. they do not coincide.

4.3.7 Timoshenko beam element models – B31

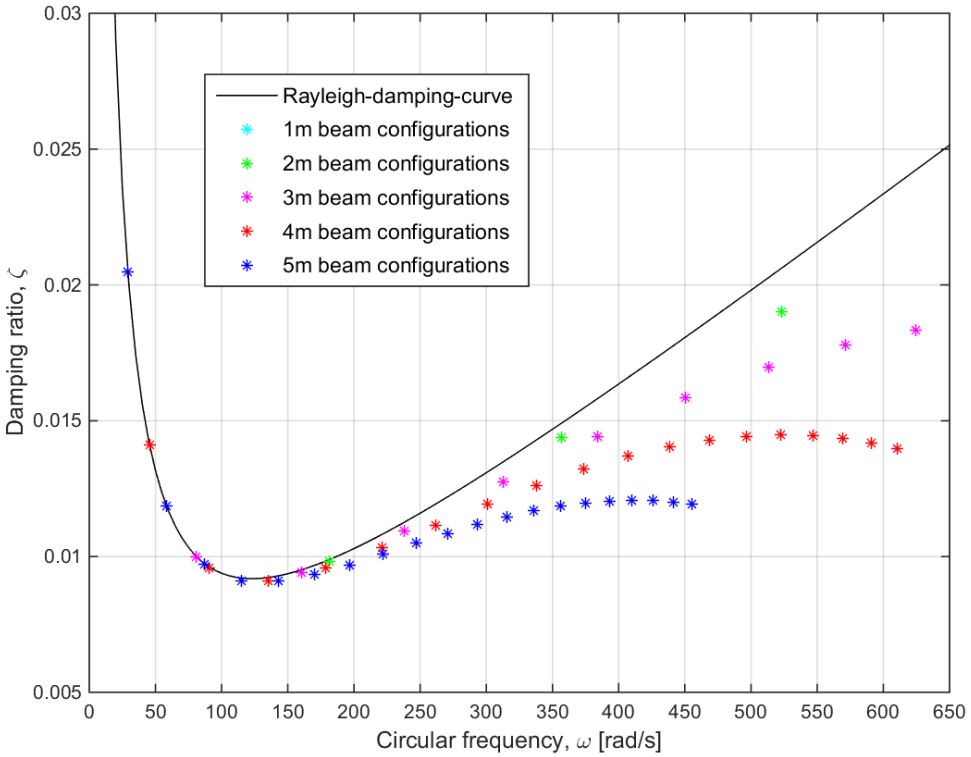

Figure 4.23: Timoshenko beam model. Damping from Abaqus plotted towards the Rayleigh-curve. First vertical mode of the beam.

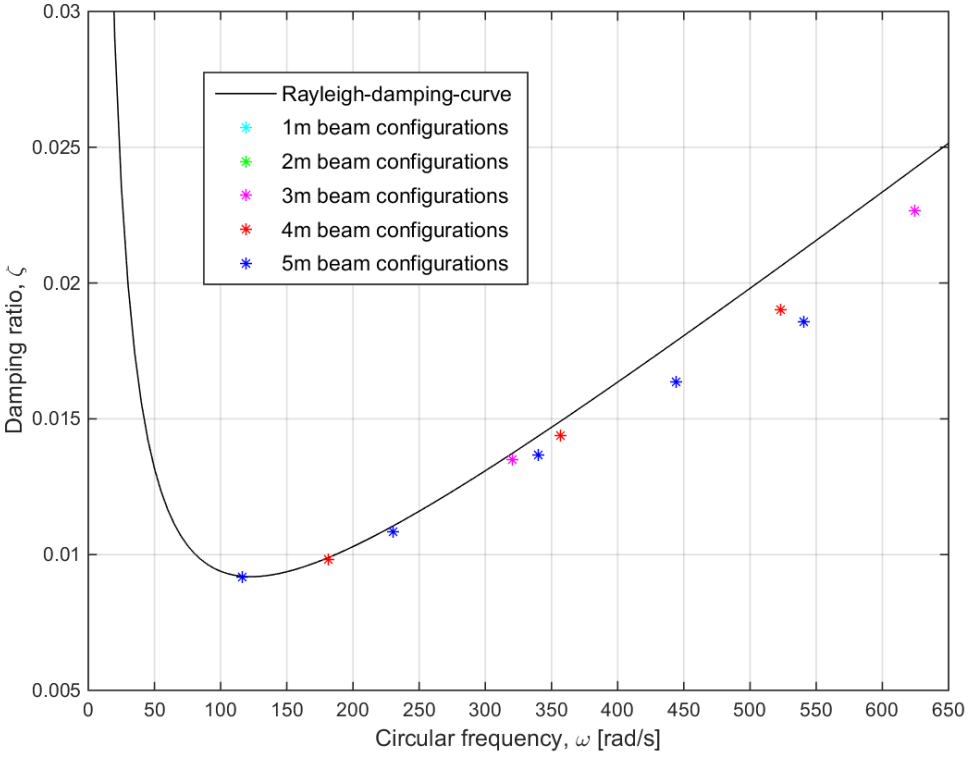

Figure 4.24:Timoshenko beam model. Damping from Abaqus plotted towards the Rayleigh-curve. Second vertical mode of the beam.

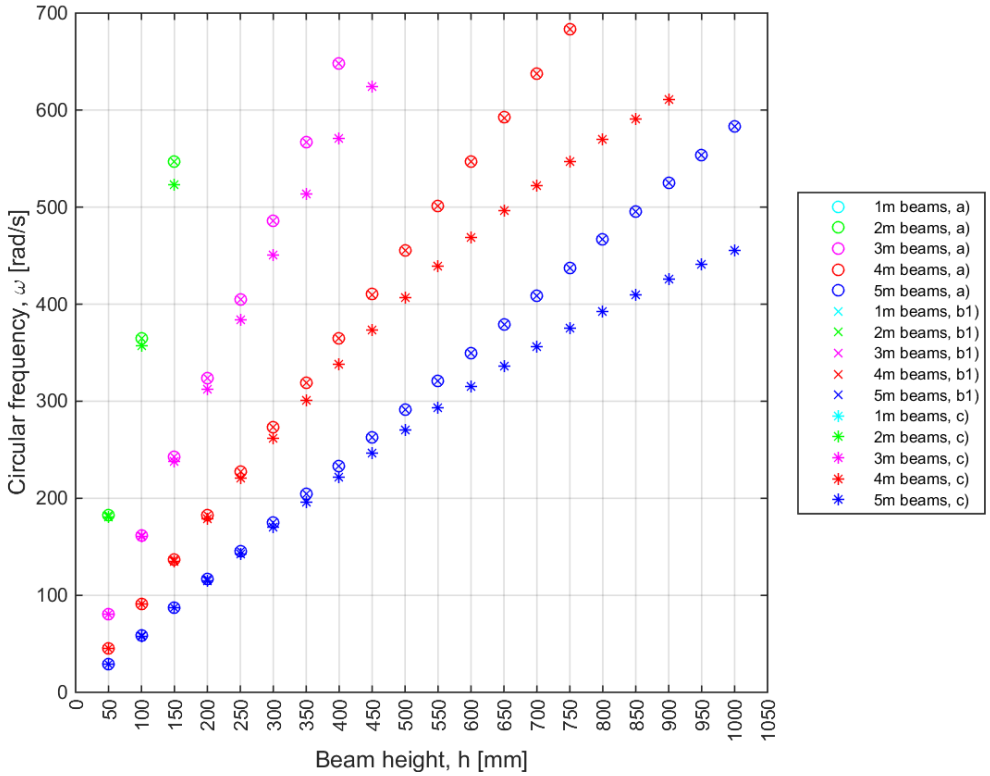

Figure 4.25:Timoshenko beam models. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Analytical damped by 2% and c) Complex frequencies from Abaqus.

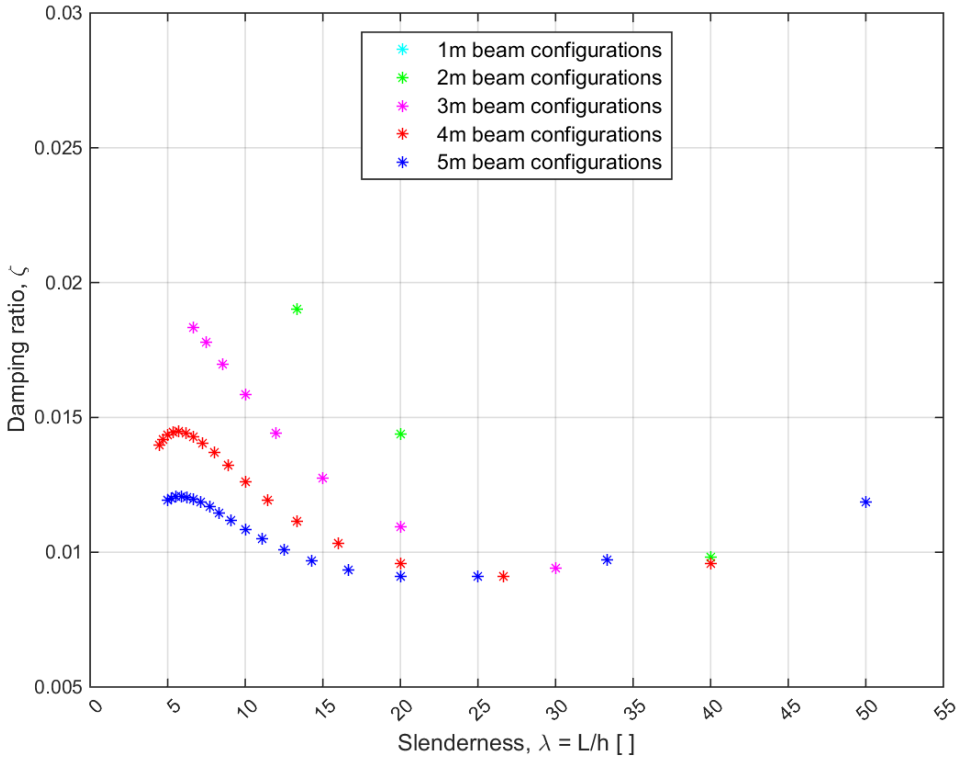

Figure 4.26: Timoshenko beam model. Damping from Abaqus plotted for the slenderness. L = span length, h = beam height.

From the first two figures, Figure 4.23 and Figure 4.24, it is seen that there is larger discrepancy between damping ratios from Abaqus and the Rayleigh-curve, increasing discrepancy for increasing eigenfrequency, especially for the first vertical mode. For eigenfrequencies below 200-250 rad/s there is good agreement, but deviations increase for frequencies above that.

Figure 4.25 confirms the discrepancy from the two preceding figures and shows that the trend also follows increasing beam height. The analytical undamped and 2%-damped eigenfrequencies are almost equal, but the complex eigenfrequencies deviate. The dimensionless plot in Figure 4.26 display similar trends between beam lengths but the curves do not coincide. The damping increase for decreasing slenderness, and has a maximum value at a slenderness of approximately 6 to 7, below which the damping ratio decreases for lower slenderness.

4.3.8 Conventional shell element models – S8R

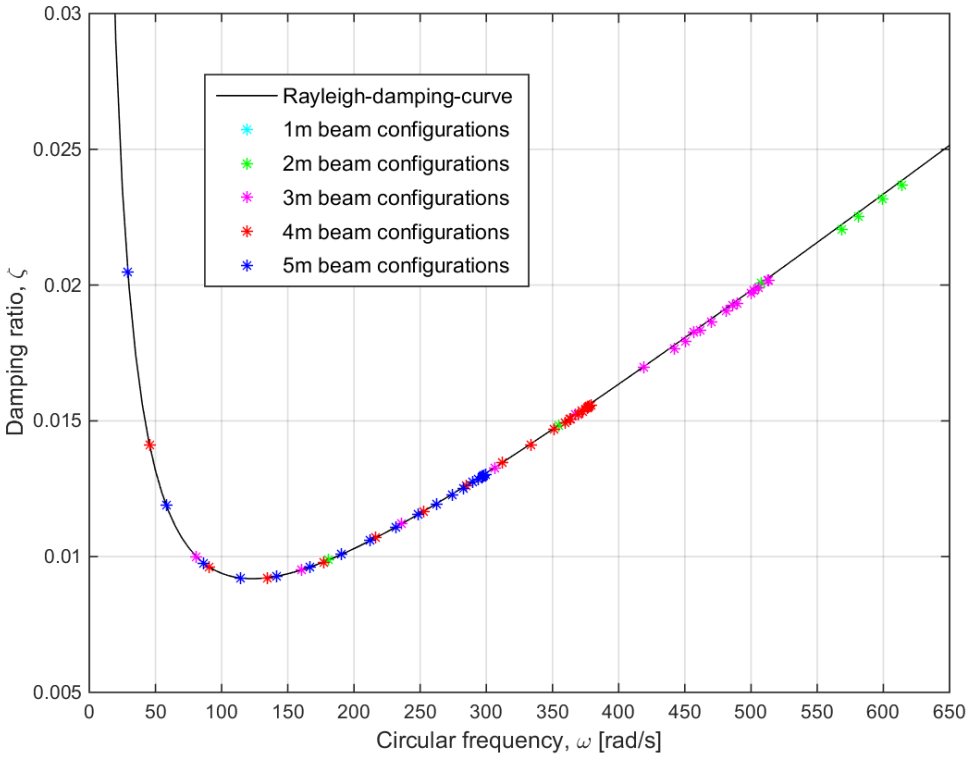

Figure 4.27: Shell model. Damping from Abaqus plotted towards the Rayleigh-curve. First vertical mode of the beam.

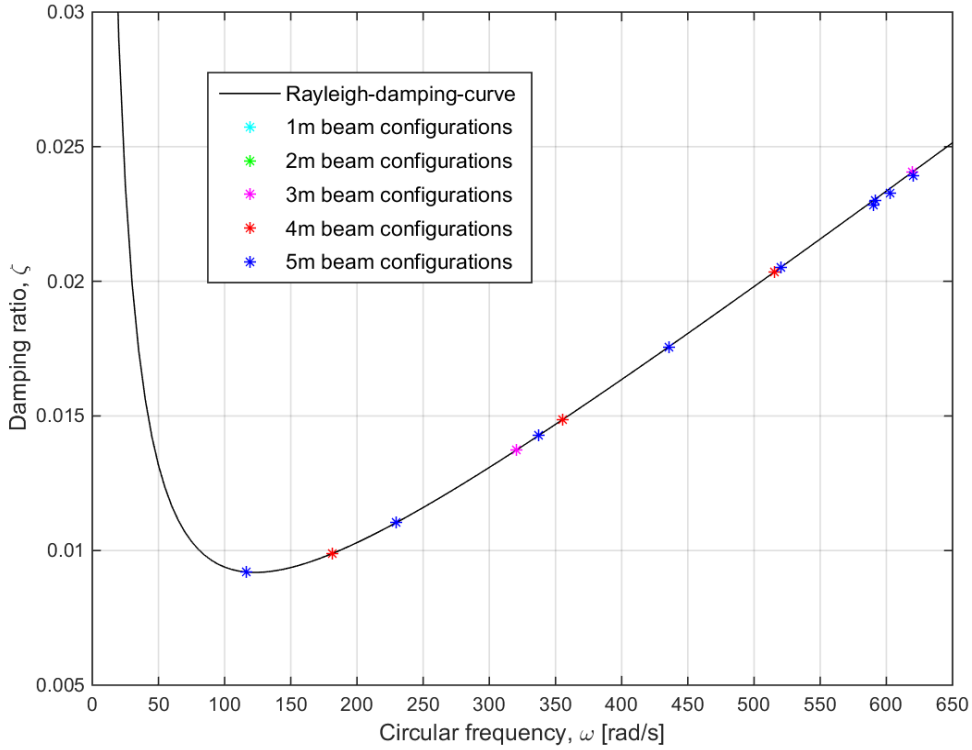

Figure 4.28: Shell model. Damping from Abaqus plotted towards the Rayleigh-curve. Second vertical mode of the beam.

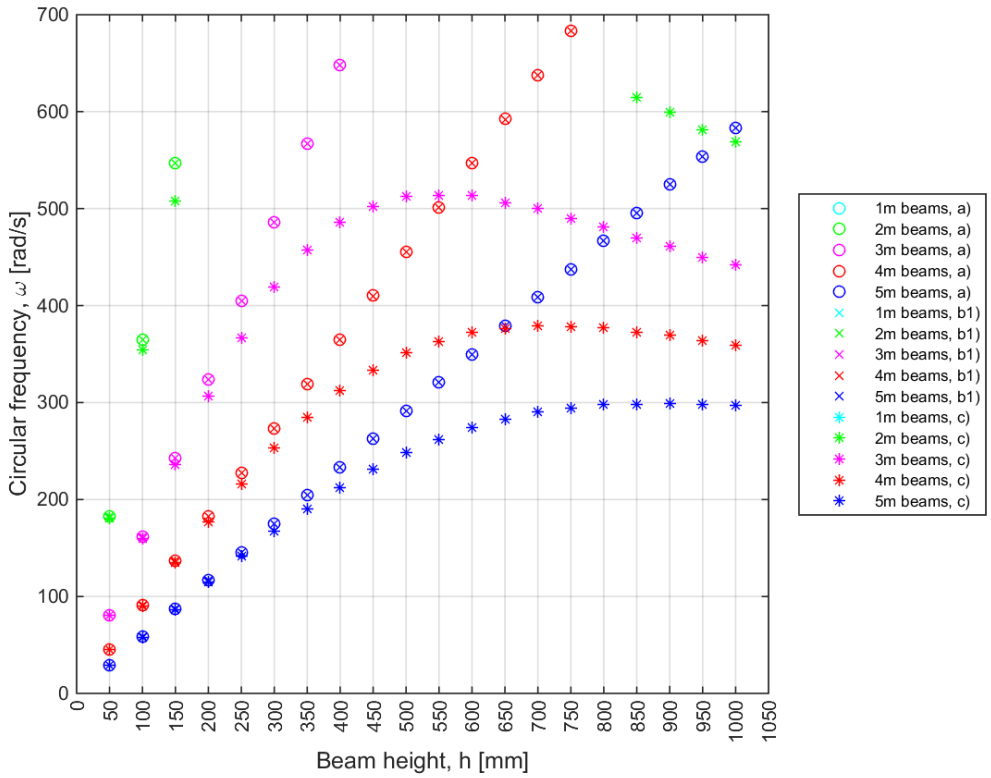

Figure 4.29: Shell model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Analytical damped by 2% and c) Complex frequencies from Abaqus.

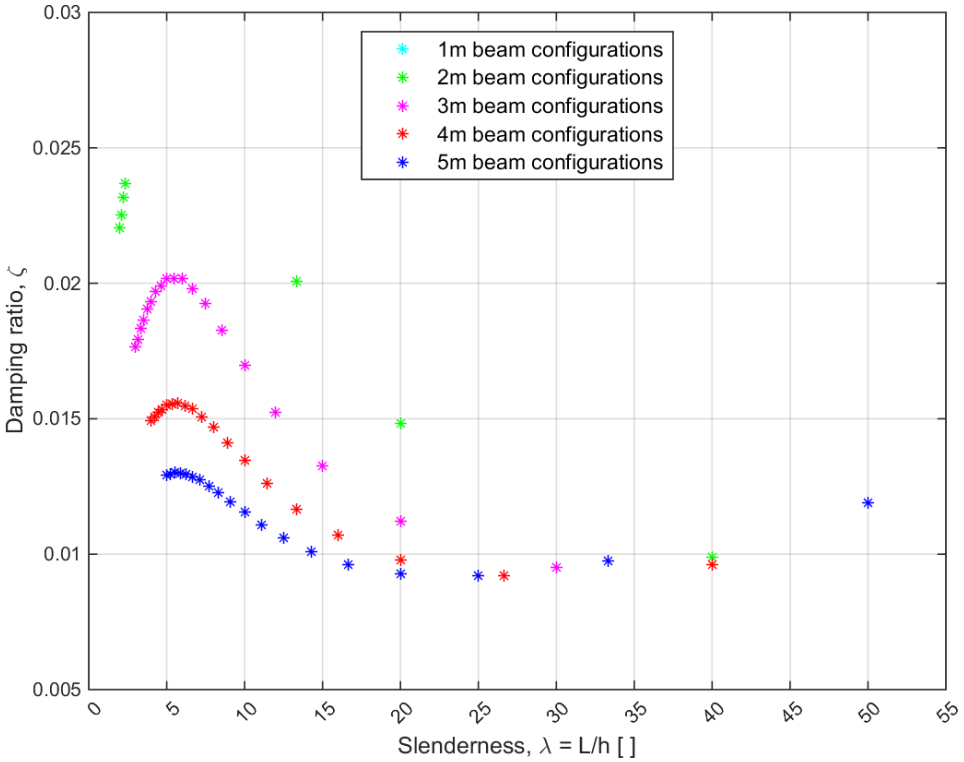

Figure 4.30: Shell model. Damping from Abaqus plotted for the slenderness. L = span length, h = beam height.

In the figures presented here, the shell models are modelled with the shell thickness representing the beam width. For the first vertical mode in Figure 4.27 and the second vertical mode in Figure 4.28, the shell model seem to be in good agreement with the Rayleigh-curve with the values from Abaqus lying at or approximately at the curve for all the frequencies. By closer inspection the result values for the highest frequencies seem to be clustering up.

The clustering seen in the latter figures are also seen in Figure 4.29. For example, for the beam length of 5 meters, the six tallest beams yield about the same eigenfrequency. Keeping this in mind and viewing Figure 4.27 and Figure 4.28 one can also see that the damping ratio for these heights must be about the same. Similar patterns can be seen for the beam lengths 3 and 4 meters.

Figure 4.30 displays similar trends between beam lengths. The damping increase for decreasing slenderness up to a maximum level, for which the slenderness is about 6. Below this slenderness, damping decreases. The decrease for the least slender beams is especially evident for the three-meter long beams.

The beams were also modelled such that the shell thickness represented the beam height. This way of modelling makes more sense for a plate model (to have the plate height as the thickness of the shell element) and the results were less reliable compared to the model presented here. For completeness, the result from the model with shell thickness as the beam height is presented in the Appendix F.

4.3.9 3D solid element models – C3D20R

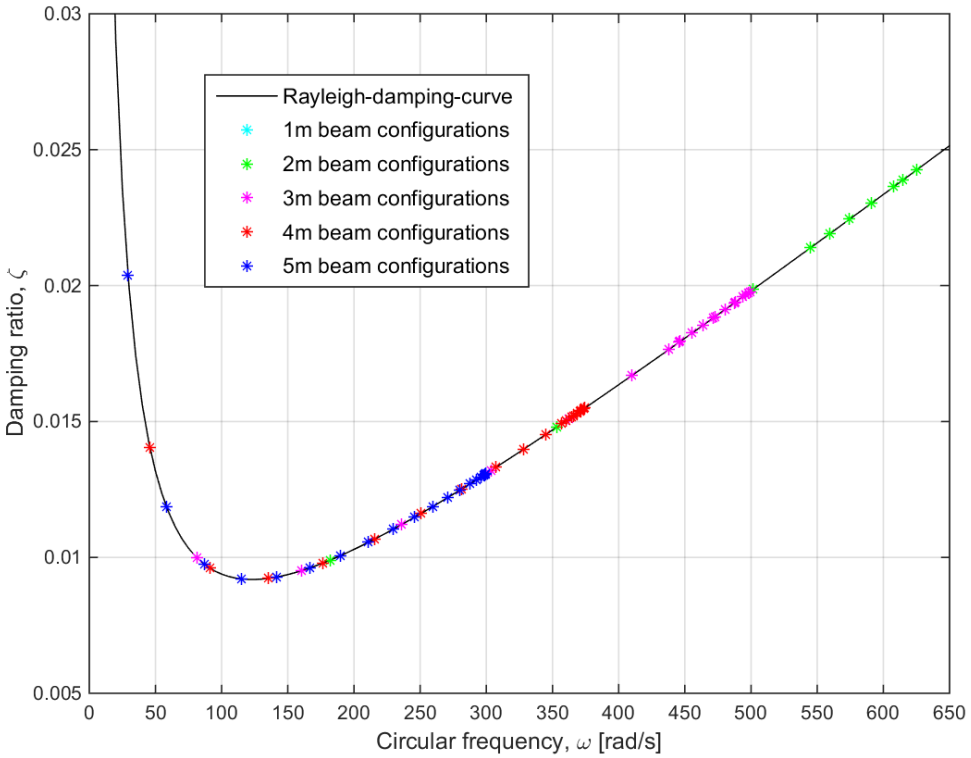

Figure 4.31: 3D solid model. Damping from Abaqus plotted towards the Rayleigh-curve. First vertical mode of the beam.

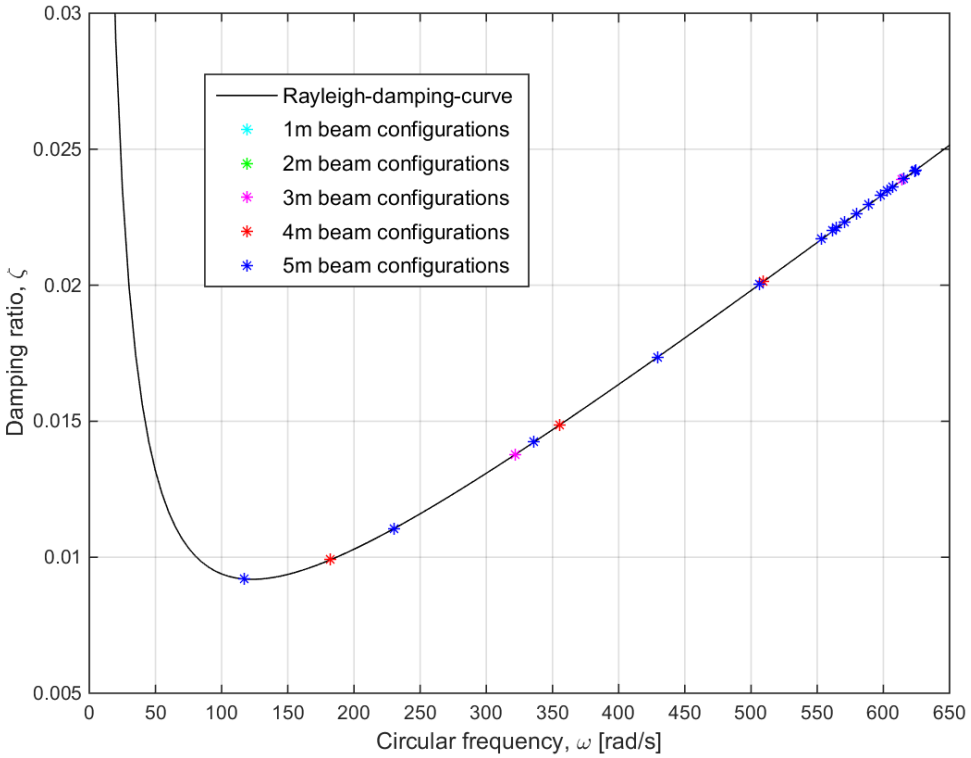

Figure 4.32: 3D solid model. Damping from Abaqus plotted towards the Rayleigh-curve. Second vertical mode of the beam.

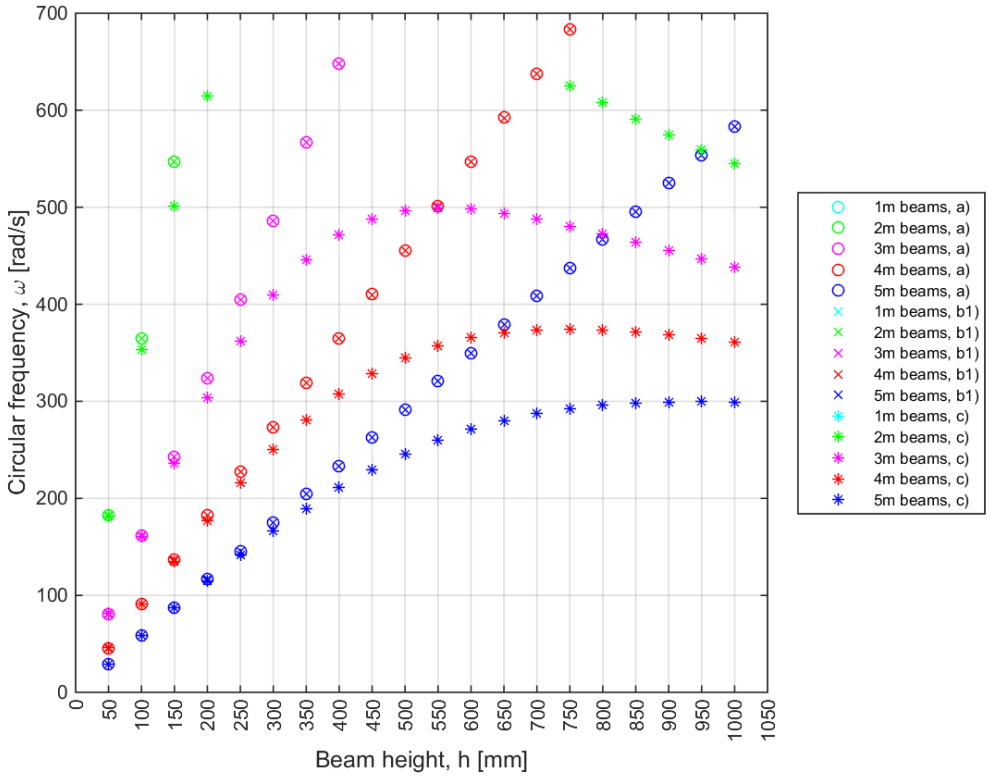

Figure 4.33: 3D solid model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Analytical damped by 2% and c) Complex frequencies from Abaqus.

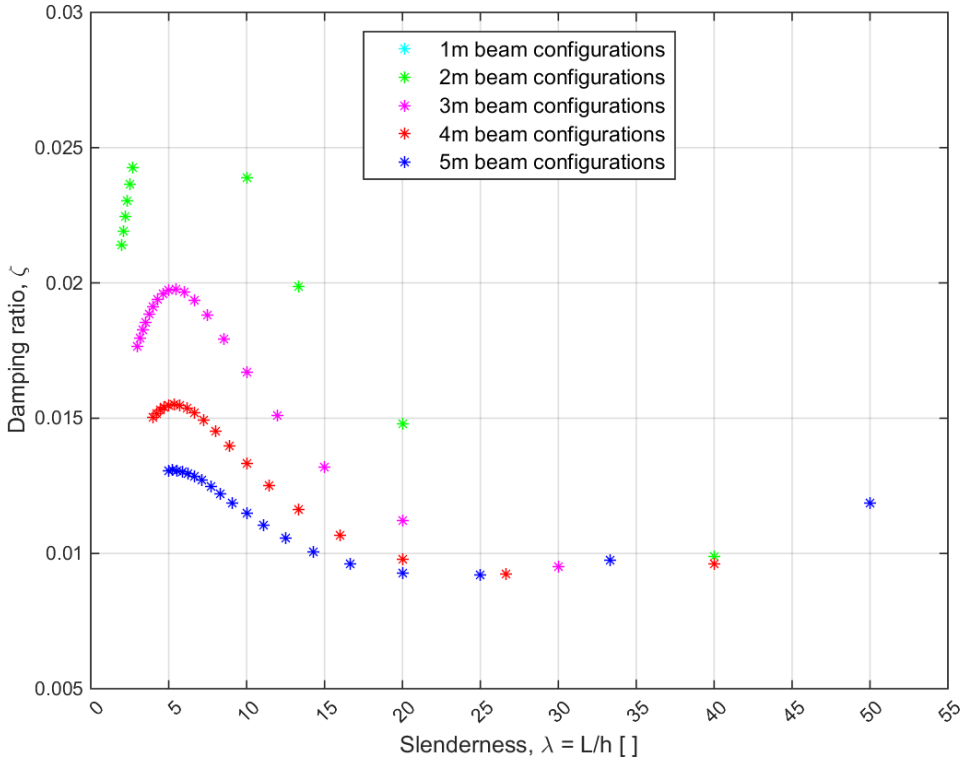

Figure 4.34: 3D solid model. Damping from Abaqus plotted for the slenderness. L = span length, h = beam height.

In Figure 4.31 and Figure 4.32, respectively for the first vertical and the second vertical mode, a clustering of data points can be seen. Further, all the data points seem to coincide perfectly with the Rayleigh-curve.

Figure 4.33 gives more insight to the clustered data points. For all beam lengths, above certain beam heights, the figure shows that eigenfrequencies decrease for increasing beam height for the beams spanning three meters, and a flattening to a specific level for the same case with the beams spanning four and five meters. Frequencies for the two-meter beams seem to exceed the 100 Hz upper limit for beam heights above 250 mm, and then decrease and reappear for beam heights of 750 mm and higher, for which the frequencies decrease with increasing beam height as for the three-meter beam.

Damping ratio plotted for the slenderness, λ , in Figure 4.34 show that different beam lengths display similar trends, but the damping ratios are in general higher for shorter beams. Additionally, all the beams have a maximum value for damping ratio about a slenderness of 5 to 6. The "missing" data points – the data points for the beams likely with eigenfrequencies above 100 Hz – for the beam spanning two meters, seem to be located about this maximum value for the damping ratio.

4.3.10 Summary of iSight results

For all element-models, except Euler-Bernoulli, either, or both, of damping ratios and eigenfrequencies deviates respectively from the mathematical Rayleigh damping model and the analytical eigenfrequencies calculated by the formula based on EB-theory. It corresponds with theory that resulting frequencies for the EB-element models coincide with the analytical formula based on the same theory. For the Timoshenko models, deviations are visible for both damping and eigenfrequencies, although the latter is expected since eigenfrequencies should be lower when shear is included. This is because effects of shear lead to a softer behaviour in the sense that additional displacements (deformations) are included. Having already considered that the eigenfrequencies are lower, the damping ratios are still too low for the corresponding frequencies.

Comparing the shell models and the 3D solid models, the damping ratios coincide with the Rayleigh curve in both cases. Thus, it seems that damping is calculated correctly for these element models. However, one also sees that a selection of the data points clusters up at some limited frequency intervals. This is also displayed in the frequency/beam height plots, which show that frequencies differ noticeably from the Euler-Bernoulli-based analytical formula, with larger differences for the higher the beams. Although the differences in frequencies are particularly evident for these models, conclusions about the accuracy of the frequencies should not be drawn from this, since these elements, like the Timoshenko elements, are shear-flexible and expectedly should not coincide with EB-theory. It should be noted that when the frequencies decrease, the damping ratios will still coincide with the Rayleigh curve as long as the damping ratios decrease in accordance with the frequencies. However, although the eigenfrequencies should not

increase linearly, as corresponds with EB-theory, the expected is still that frequencies should increase for increasing beam heights, which for the highest beams is not the case here.

In many cases, trends appear to be more evident when parameters are plotted as dimensionless quantities. Therefore, the damping ratios were plotted for the dimensionless parameter, beam slenderness, λ. The damping ratio/slenderness plots show that different beam lengths display similar trends for the damping ratio, but they do not coincide. Thus, there is no simple relationship between damping ratio and slenderness.

Further, the deviations in damping ratios are confirmed by the figures with damping ratio/slenderness plots. For clarity, a remark could be made that the least slender beams correspond to the highest beam cross-sections. The Euler-Bernoulli models behave as expected: reducing the slenderness leads to increased damping ratios. This is expected since Rayleigh damping is implemented. If one review the Rayleigh-curve and the damping ratio/slenderness plot, one can see that the shapes match each other. They match since the most slender beams are not so stiff and should have low eigenfrequencies and then high damping ratios. Moving towards less slender beams the beams become stiffer and the damping ratio should decrease before it starts to increase. The increase in damping ratio brings the beams into the stiffness dominated region of the Rayleigh damping model.

However, there is an exception that is common for the Timoshenko, shell, and 3D solid model. There is a dip in damping ratio for the least slender, i.e. the thickest, beams. This dip occurs for all the said beams below a slenderness about 5 to 6. At these values of slenderness, the maximum level of damping occurs, and for slenderness values reduced below these values, the damping ratio decreases.

The most evident difference between these three elements and the EB-element, is that they all include shear deformations while EB do not. It seems from this that when using Rayleigh damping together with shear-flexible elements (in Abaqus), damping ratios are underestimated according to the Rayleigh damping model for slenderness (λ) values below approximately 5 to 6.

It is interesting that the deviations from the Rayleigh damping model occur for all the beams with elements that include shear deformation when their slenderness indicate that shear deformation would play a significant role. If it was shear that caused this, one could expect to find it also for the second vertical modes of the beams. When viewing the second vertical mode of the Timoshenko beam there are several data points in the stiffness-dominated region of the Rayleigh model, but these do not deviate that much from the Rayleigh curve. These data points correspond to quite slender beams, so shear deformations are in that regard not that important, but the second vertical mode induce more shear in the beam than the first vertical mode. These observations indicate that

there is not a simple connection between deviations in damping ratio and shear in the beams.

4.4 Further review of iSight-results

The above results for all the models, except from the Euler-Bernoulli model, was unexpected. The results were expected to match the Rayleigh-curve, and three of the models did not. Additionally, they mismatched in a similar pattern. This was tested manually in Abaqus with several sample tests and all of them gave the exact same results as from the iSight-simulations, which suggest that the deviations are not caused by iSight.

These results are extracted from the complex frequency step. To check for problems with the solution in this step, a test with an alternative measurement of damping was done. This alternative was the logarithmic decrement method, which is presented in section 2.3.

The method was applied by setting a prescribed deformation to the top of the beam in the middle of the span. This deformation was "deactivated" in a time integration step ("Dynamic, Implicit") and the resulting free vibration behaviour of the middle point was plotted in Abaqus. This decaying motion was then analysed with the logarithmic decrement method to get the damping.

By running this analysis on sample tests of some model configurations that yielded discrepancies, the results from the sample tests were checked. Beam configurations that showed unexpected results for the Timoshenko beam model, the shell model and the 3D solid model was checked. The results from the check is presented in Table 4.2.

Table 4.2: Results from two methods for measuring damping. B31 is the Timoshenko beam, S8R the shell model, C3D20R the 3D solid model, L = span length, h = beam height. Time integration is the Dynamic, Implicit step measured by the logarithmic decrement method.

	Damping ratio	
	Complex frequency step	Time integration
B31, L = 4 m, h = 650 mm	0.014	0.013
S8R, 5 m, h = 600 mm	0.012	0.012
C3D20R, 3 m, h = 350 mm	0.018	0.017

Table 4.2 shows that the damping ratio from the time integration and the complex frequency step is very close to each other. These are considered so close to one another that problems with the solution from the complex frequency step does not seem to explain the unexpected results from iSight.

4.4.1 Increasing the shear stiffness

The iSight simulations show that the Euler-Bernoulli (EB) beam model reports damping ratios that coincide with the Rayleigh-curve. This means that one can predict which damping levels will occur for the different eigenfrequencies with this model. For the models with Timoshenko beam theory, Mindlin-Reissner shell theory and 3D solid theory the lower eigenfrequencies are consistent with the Rayleigh-curve, but they deviate for the higher eigenfrequencies – larger deviation the higher the frequency. One property that separate EB from the rest is the omission of shear stiffness. The EB model can be interpreted as infinitely stiff in shear, whereas the other models are softer. In other words, when being sheared, the cross section will, to some extent, rotate additionally due to the shear.

To investigate if this could affect the results, an increase of the different shear stiffnesses was tested. Initially, the knowledge on the EB theory was tested by increasing the stiffness for the three shear moduli one at the time. This had no effect on the damping ratio or eigenfrequency. This was as expected and strengthens the understanding of the EB model.

Next, a beam configuration of 3 m span length and 300 mm beam height was tested more extensively. This beam has a height to length ratio (or aspect ratio, h/L) of 1/10, which is commonly considered to be the transition between recommendations for when EB and Timoshenko theory should be applied. The first vertical mode of this beam was reported at about 70 Hz. This beam was chosen since it was assumed that the beam would be high enough for Timoshenko beam theory to have an effect, meaning that beams with a very low aspect ratio is practically unaffected of shear effects [17]. Further, the beam has a high value for the eigenfrequency of the first mode. In the analysis there is set a limit at 100 Hz for the eigenfrequencies, so a taller beam was avoided in case adjustments made would send the eigenfrequency above 100 Hz.

This beam was tested for Timoshenko beam theory (B31-elements) and Mindlin-Reissner shell theory (S8R elements). For the shell model the shell thickness represented the beam width, as this seemingly displayed the most reliable results. The original value for the shear modulus perpendicular to the fibre direction (G23) is 30 MPa; shear in this direction is commonly referred to as rolling shear. This was tested with values ranging from 10 to 200 MPa.

As described in the previous paragraph, two different models were tested but only results from the shell model are presented here. The behaviour for the other model, the beam model, was similar. Figure 4.35 show that changes in rolling shear stiffness did not affect the damping ratio of the first vertical mode, this was also true for the eigenfrequency.

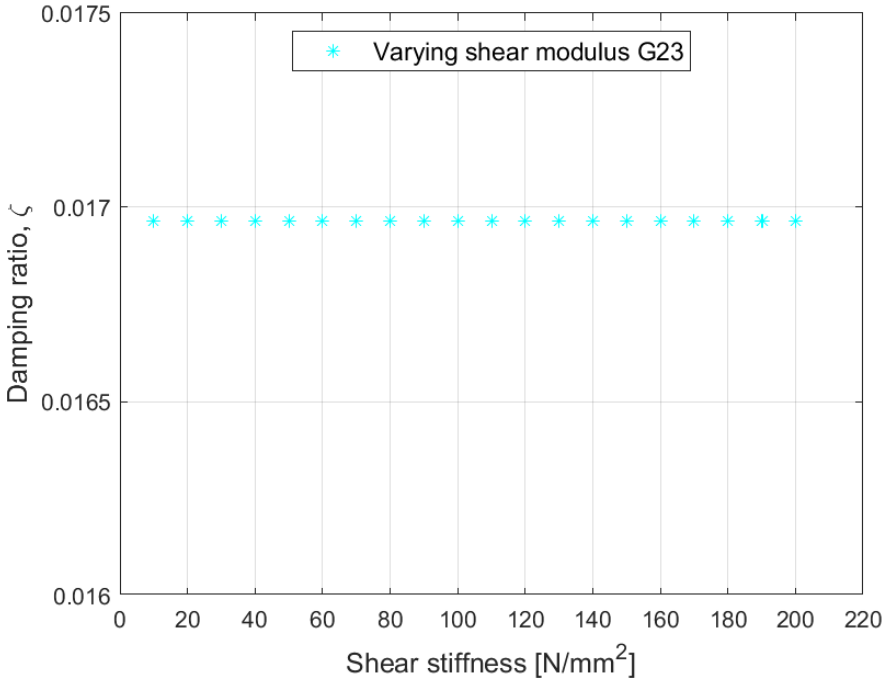

Figure 4.35: Damping ratios for a shell model with varying shear modulus G23 (rolling shear).

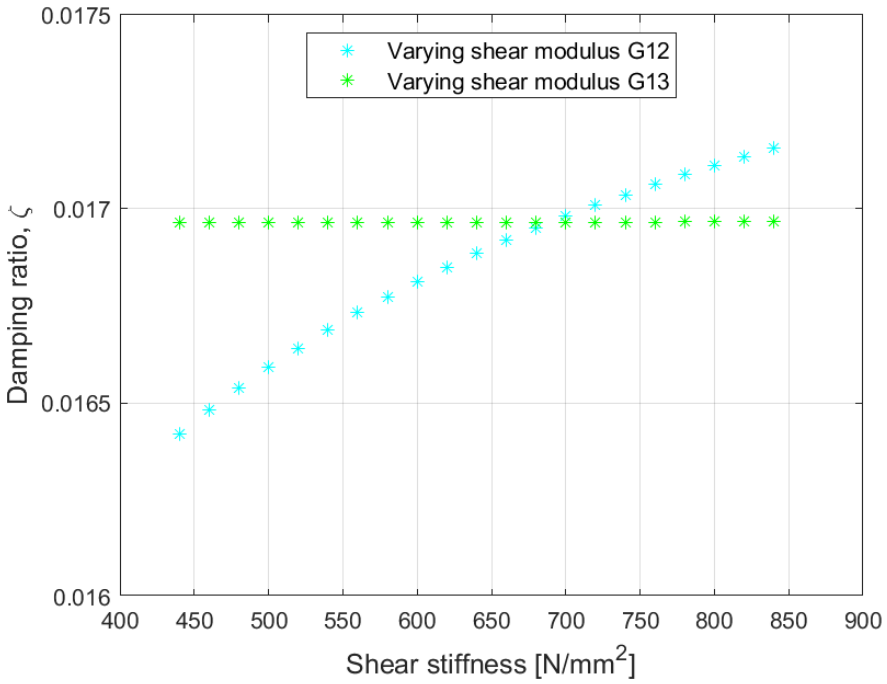

Figure 4.36: Damping ratios for a shell model with one shear modulus varying at the time. Done for G12 and G13.

When changing the shear stiffnesses parallel to grain, G12 and G13, one was held constant while the other was tested in the range 440 MPa to 840 MPa. The original value for both moduli were 690 MPa. Figure 4.36 shows how the damping ratio is unaffected by changes in the shear modulus G13 but has a small increase with increasing shear

modulus G12. Note how short a range the vertical axis is spanning, the changes in damping ratio are relatively small. Generally, changes in G12 affected the vertical modes and did nothing to the horizontal modes. For G13 it was opposite.

These results indicate that for the first vertical or horizontal mode, the shear modulus G23 does not have an impact on neither damping ratio nor eigenfrequency. The shear moduli G12 and G13 has an impact, one at the time, dependent upon the direction that the mode is vibrating in.

4.4.2 Replicating the Euler-Bernoulli theory by other theory

In the analysis of different geometrical configurations, the Euler-Bernoulli (EB) beam matched the Rayleigh-curve and yielded expected eigenfrequencies. By this, the EB beam can be viewed as stable and predictable. Along with the knowledge from the previous section, on how the shear moduli affect different types of vibration modes, the Timoshenko beam and the shell model were subject to further investigation. As explained in section 4.4.1 the EB beam differ from the Timoshenko beam and shell model as it can be thought of as infinitely stiff in shear, since it does not include shear deformations.

The stable and predictable behaviour seen from the EB beam is wanted and is sought replicated by mimicking these beams. This was done by setting the shear modulus much higher, to 6 900 MPa, which is 10 times the original value. Then a comparison was made between the beams with the original shear stiffness and the increased shear stiffness.

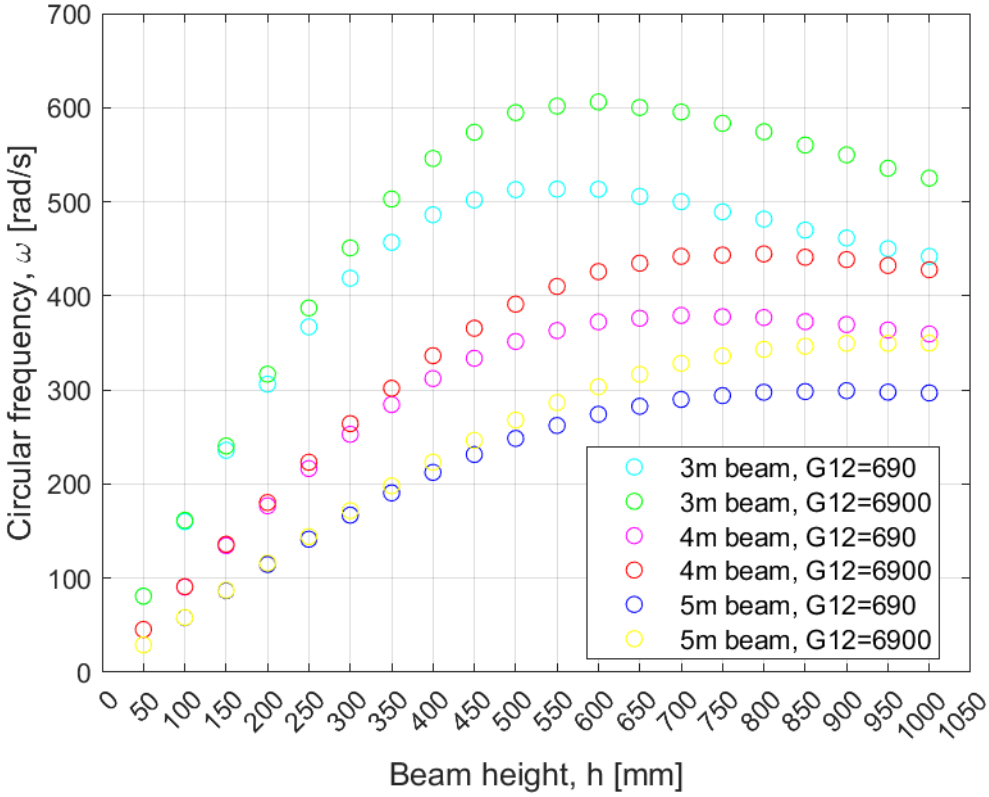

Figure 4.37: Shell model. Eigenfrequencies for the first vertical mode for different beam heights and span lengths. All results are for G12, the shear modulus, as 690 MPa and 6 900 MPa.

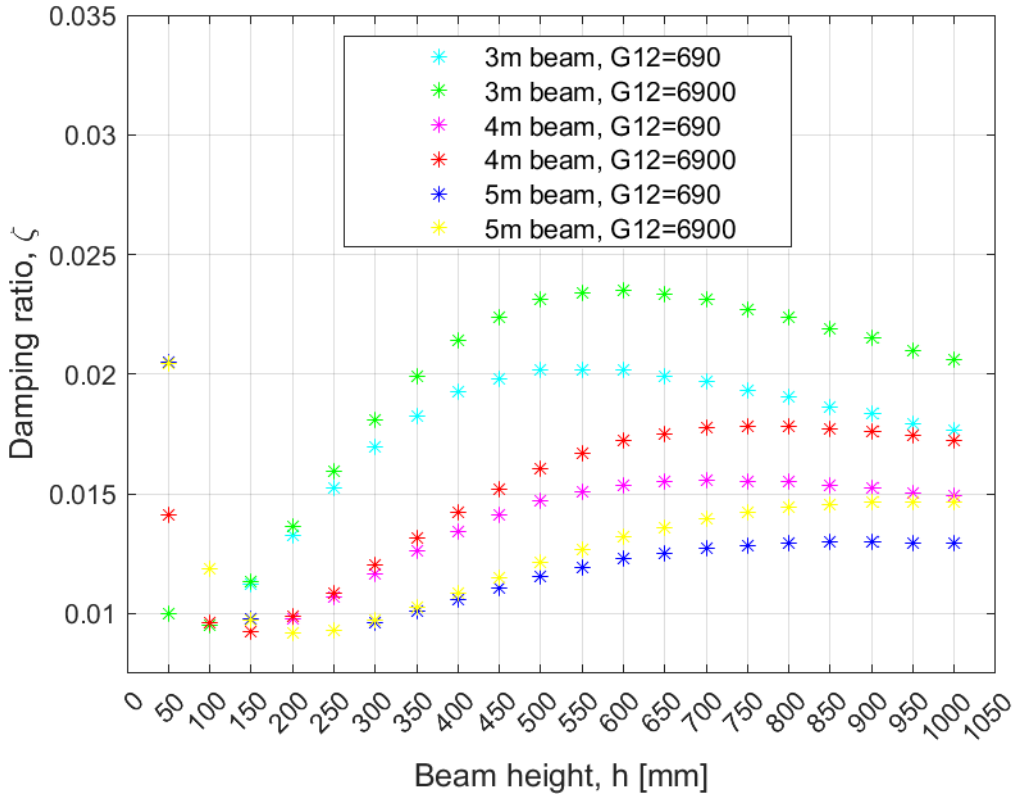

Figure 4.38: Shell model. Damping ratios for the first vertical mode for different beam heights and span lengths. All results are for G12, the shear modulus, as 690 MPa and 6 900 MPa.

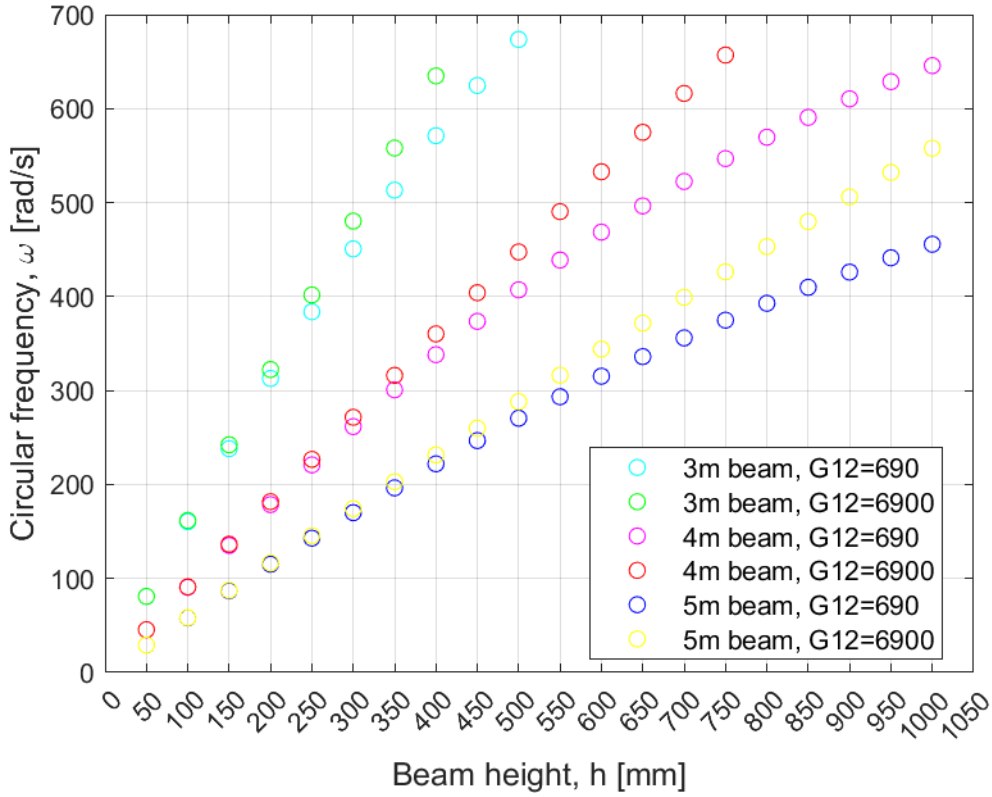

Figure 4.39: Timoshenko beam model. Eigenfrequencies for the first vertical mode for different beam heights and span lengths. All results are for G12, the shear modulus, as 690 MPa and 6 900 MPa.

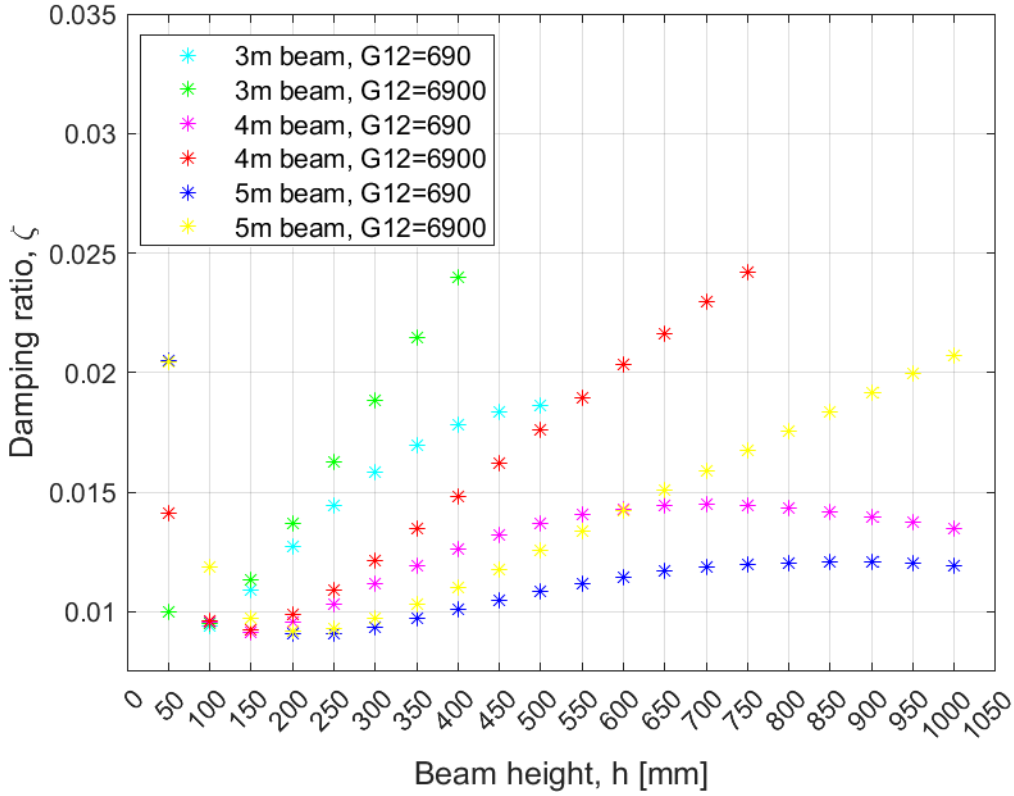

Figure 4.40: Timoshenko beam model. Damping ratios for the first vertical mode for different beam heights and span lengths. All results are for G12, the shear modulus, as 690 MPa and 6 900 MPa.

Figure 4.37 and Figure 4.38 respectively show how eigenfrequency and damping ratio is affected by the increase in shear stiffness for the shell model. The plots show similar trends for both the original and the increased shear stiffness. For the lowest beams, i.e. the most slender beams (highest λ-value), the results are unaffected by increasing shear. As the beam heights increase, the increased shear stiffness has a relatively larger influence and the beams with increased shear stiffness reach higher eigenfrequencies and damping ratios.

Further, the shape made up from the plotted data points are similar for the two cases. There is a maximum value for eigenfrequency and damping ratio, and as the beam height is further increased, both the eigenfrequency and damping ratio decline. The exception is for the beams spanning 5 m, for which the trend curves for frequency and damping ratio flattens out instead of decreasing, but one may assume that these would also decline if the beam height had been increased further, exceeding 1 000 mm. The general difference between normal and increased shear stiffness for all beam lengths, is that the maximum values are larger and occur for taller beams in the latter case, as compared to the beams with a shear stiffness of 690 MPa.

Results for the Timoshenko beam are presented in Figure 4.39 and Figure 4.40. Figure 4.39 show the eigenfrequencies for the different beam configurations and Figure 4.40 show the damping ratios. For the damping ratios, the beams with the original stiffness behave like the shell model but the ones with increased shear stiffness do not. Those with increased shear stiffness have a seemingly linear increase in damping ratio from about 150-250 mm beam height, dependent upon span length. For the lowest beam heights, the eigenfrequencies are low and according to the Rayleigh damping model this will give high damping ratios. As seen for the original shear stiffness, the damping ratio then decrease for increasing beam height, further starts to increase, flatten out and decrease.

The eigenfrequencies in Figure 4.39 are also affected by the increase in shear stiffness. For the original shear stiffness there is only a mild tendency of decay in the increase of eigenfrequency with beam height. With the increased shear stiffness this decaying does not occur and the eigenfrequency keep on increasing linearly.

This has shown that none of the beams displayed predictable results with respect to the mathematical Rayleigh-curve, such as the Euler-Bernoulli beam. For the shell model it seems like the same trend from the 690 MPa case is also occurring for the 6 900MPa case, only with a shift towards greater beam heights and, with it, higher eigenfrequencies. Timoshenko beams show a linear increase for the eigenfrequency when applying increased shear stiffness, but not for the damping ratios. The eigenfrequencies behave more like the EB beam. The effect of the increased shear stiffness is greatest for the thickest beams. These are the beams where shear effects play a more important role, so this is not surprising but worth noting.

4.4.3 Evaluation of Abaqus eigenfrequency calculation

An attempt was made to evaluate the eigenfrequency calculations in Abaqus for the Timoshenko element models. This was done by solving the eigenvalue problem analytically, respectively with two and five Euler-Bernoulli (EB) beam elements and with two and five Timoshenko (T) beam elements (shear-flexible element formulation), and comparing the resulting eigenfrequencies to those calculated by the analytical formula based on Euler-Bernoulli theory and to those calculated by Abaqus for the Timoshenko B31 beam element models. The MATLAB script used for the analytical calculations are included in Appendix B.

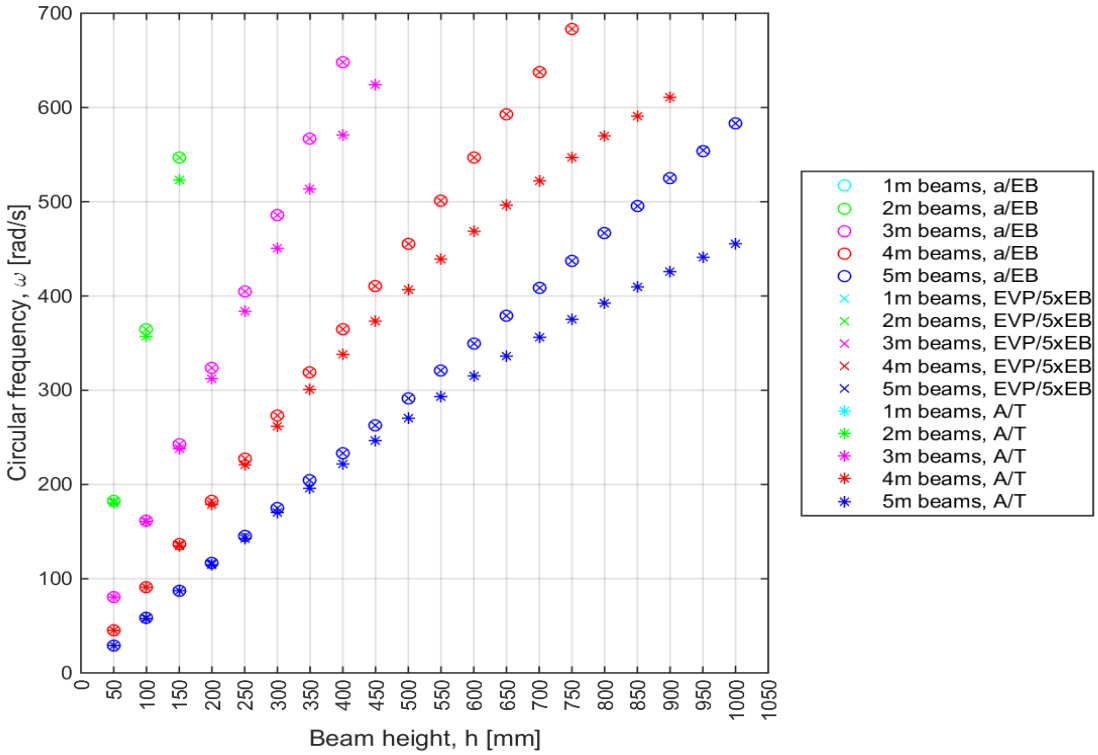

Figure 4.41: Frequencies plotted for increasing beam heights. a/EB) Eigenfrequencies calculated by the analytical formula based on Euler-Bernoulli theory. EVP/5xEB) Eigenfrequencies calculated from eigenvalue problem with 5xEB beam elements. A/T) Eigenfrequencies calculated in Abaqus models with Timoshenko B31-beam elements.

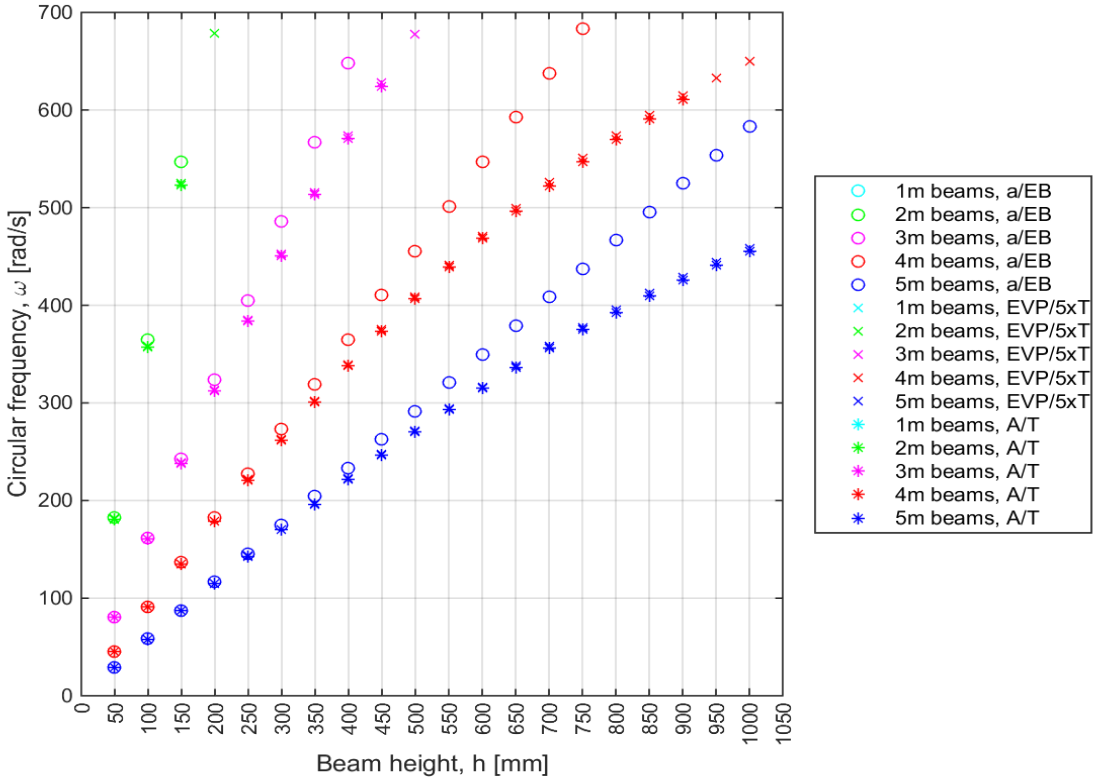

Figure 4.42: Frequencies plotted for increasing beam heights. a/EB) Eigenfrequencies calculated by the analytical formula based on Euler-Bernoulli theory. EVP/5xT) Eigenfrequencies calculated from eigenvalue problem with 5xT beam elements. A/T) Eigenfrequencies calculated in Abaqus models with Timoshenko B31-beam elements.

In section 4.3.6, Figure 4.21 showed that the eigenfrequencies calculated by Abaqus for the Euler-Bernoulli B33-beam elements coincided with the frequencies calculated by the analytical EB-based formula presented in Equation (2-9) in the theory section. Expectedly, when solving the eigenvalue problem analytically with EB elements, the solutions should thus also converge towards the solutions from the analytical formula. This is verified by the results plotted in Figure 4.42, which shows that the resulting eigenfrequencies from solving the EVP with five EB beam elements have converged to the analytical EB eigenfrequencies. With only two EB elements, convergence was not fully achieved, although the improvement when increasing to five elements indicate that when the number of elements is adequate, Abaqus solutions with B33-elements converges towards the analytical EB theory solutions.

Similarly, when solving the EVP analytically with Timoshenko beam elements, the solutions should converge towards the Timoshenko B31-element models in Abaqus when damping levels are low. Eigenfrequencies calculated from the EVP with only two Timoshenko beam elements forming the system, does not coincide with the Abaqus solutions. However, as displayed in Figure 4.42, when increasing to five elements the solutions seem to converge towards those from Abaqus. Although full convergence is still not achieved, indications are strong that this will be the case when increasing the number of elements to the corresponding of the numerical models.

It should also be noted that in this frequency evaluation, only the undamped EVP was solved. However, as mentioned in previous chapters, for low levels of damping the differences between damped and undamped frequencies are negligible or practically zero. Thus, the results seem to verify that Abaqus frequency calculations coincide with the expected for both Euler-Bernoulli and Timoshenko beam theory.

Further, considering the damping ratio plot in Figure 4.24 and frequency plot in Figure 4.25 in section 4.3.7, this indicates that the frequency calculations in Abaqus are not the cause for the deviations in damping ratios from the Rayleigh curve for the Timoshenko B31-beam element models.

Since 3D solid elements and shell elements, like the Timoshenko elements, are shear-flexible elements, one might presume that the frequency calculations for these elements will converge towards Timoshenko theory solutions as well. However, even though damping for these element models coincide with the Rayleigh curve, the frequencies increase at a relatively much lower rate than for the Timoshenko element models, and for the least slender beams, the frequencies even decrease. In this sense they seem to be more shear-flexible or have relatively more "shear-dominated" behaviour than the Timoshenko element, for which obvious causes are not evident. If so, they underestimate the eigenfrequencies for thick beams according to Timoshenko theory.

Frequency calculations of these elements should therefore have been investigated further, but geometry properties for these elements make analytical calculations much more cumbersome and such have not been performed here.

4.5 MAC-analysis

To investigate possible causes for the apparent deviations present for the beam models – between damping ratios from Abaqus models compared to the Rayleigh reference curve – MAC-analyses (modal assurance criterion) were performed on a selection of beam configurations. Although some Abaqus test models were made to investigate mode shapes directly in the Abaqus results viewport, a MAC analysis was considered as more accurate, since it may discover numerical deviations between compared mode shapes.

One intention was to check that eigenvectors, i.e. mode shapes, were not calculated incorrectly in Abaqus during the iSight-loops, in which the beam height was increased iteratively while each respective beam length was held constant. If large discrepancies in modes shapes were found for cases where it was not expected, it could imply undiscovered errors occurring due to iterative changes in the Abaqus models. For instance, imagined errors could be related to meshing procedures or automated regeneration of new beam configurations. For this purpose, investigation of mode shapes represented by vertical degrees-of-freedom seemed sufficient.

Another intentional use of MAC-analysis was to reveal shear effects, meaning shear deformations (or shear angle), and investigate the influence of shear on the mode shapes for specific ranges of beam configurations, and thus evaluate whether shear effects could be the cause of deviating results. The eventual presence of shear governed mode shapes – expectedly for high, short beams with low slenderness, λ – could possibly be found by extracting and comparing mode shapes for different beam heights, based on respectively displacements or rotations in degrees of freedom corresponding to respective modes.

Euler-Bernoulli beam elements (EB-elements) do not take shear deformations into account and may display only bending modes. Thus, the mode shapes of different EB-element models should be only scales of one another's modes. Since MAC-values are not sensitive to, and in fact not affected by, the scaling of mode shapes, MAC-values calculated for comparison of any EB-models should be equal (or approximately equal, depending on numerical noise) to 1, indicating (or confirming) 100% correlation between modes.

For Timoshenko elements, 3D solid elements or other elements that can represent shear, beam models with high levels of shear deformations present compared towards beam models with lower (or no) levels of shear could then be revealed by non-correlating modes. Physically, when shear is present in a beam, the shear angle will increase, so that the beam cross section is no longer perpendicular to the beam neutral axis. In other words, the beam rotation about the out-of-plane axis is influenced by shear. Axial deformations are thus also influenced, especially at top and bottom surfaces of the beam – since axial deformations due to shear will be larger vertically far from the neutral axis of the beam.

In the cases presented here, more shear deformations are expected to be present for beam configurations of lower slenderness – where the influence of shear deformations compared to bending is relatively higher. However, non-correlated modes in MAC-analysis would presumably only be found when shear effects in a beam are present to a degree so that the mode shape is no longer neither a scaling, nor an approximate scaling, of a mode for another beam in which another level of shear (most evidently if no shear) is present. In case of small shear deformations, or little difference in shear and bending contributions between compared beam mode shapes, it will probably not be captured by a MAC-calculation.

Most samples were selected so that for one beam length configuration, different beam heights were compared towards one another, thus comparing beams of different slenderness. Euler-Bernoulli versus Timoshenko element models were investigated – since one element can and the other cannot display shear – by comparing identical beam configurations for the two element types. Comparing beam element models towards 3D solid or shell models was not possible due to different mesh configurations between models and followingly lack of compatibility between coordinates for the degrees-of-

freedom. Only vertical modes were investigated, since horizontal modes generally did not display the same levels of discrepancies as the vertical.

In all Abaqus models, except the shell element models with shell thickness as beam height, the beams were oriented so that axis 1 was along the beam neutral axis, axis 2 was the vertical axis, and axis 3 was the out-of-plane axis. Degrees-of-freedom in Abaqus corresponds to these axes so that DOF U1, U2 and U3 are translations along the respective axes, and UR1, UR2, and UR3 are rotations about the respective axes. For the shell element models with shell thickness as beam height, axis 2 was the out-of-plane axis and axis 3 was the vertical axis, while axis one remained equivalent as for the other element models.

Mode shapes (eigenvectors) may be extracted from any of the degrees of freedom for which displacements and/or rotations are calculated for the respective elements. Beam elements and shell elements have both translational and rotational DOFs. 3D solid elements only have translational degrees-of-freedom. DOFs in all nodes of respective beam planes were used, thus reducing the probability of erroneous MAC-values due to inappropriate DOF-selection. Some AutoMAC-tests – in which a mode is compared towards itself – were performed to check this, as well as validating the implementation of MAC.

The mode shape parameters were extracted from Abaqus by a package of MATLAB and Python scripts developed and provided by Øyvind W. Petersen at the Department of Structural Engineering. MATLAB scripts for further processing of the extracted data and execution of MAC-analysis were developed by the authors and are provided in Appendix C.

When the complex frequency step is used, Abaqus calculates the complex eigenvectors – i.e. both the real and imaginary parts of the eigenvectors – to the system. When the only damping included in the numerical model is Rayleigh damping, the imaginary parts of the eigenvectors are per definition 0. This is because Rayleigh damping is based on using the undamped eigenvectors, which are real, for the modal transformation of the system. The imaginary parts were zero in all cases, thus, resulting complex MAC-values were all identical to the MAC-values calculated with real parts of the eigenvectors only. Thus, discrepancies are seemingly not related to calculation of complex eigenvectors.

4.5.1 Euler-Bernoulli (B33) and Timoshenko (B31) beam element models

Both the Euler-Bernoulli and the Timoshenko beam element only have degrees-of-freedom along one dimension, i.e. along the neutral axis of the beam. Thus, eventual shear effects in beam elements may only be found through rotational degrees-of-freedom along the neutral axis. Some tests were still performed for axial mode shapes. This revealed that axial displacements along the neutral axis of the beam element models were practically zero, having an order of magnitude less than 10^{-10}. In other

words, there are no axial deformations in the mode shapes of any of the beam-element models.

All tests of mode shapes based on vertical degrees of freedom had MAC-values equal to 1, thus showing 100% correlation between modes, as expected. These results are therefore not included.

MAC-values for comparison of rotational degrees of freedom for the first vertical mode shape of different beam configurations and element types are presented in Table 4.3.

Table 4.3: MAC-values for comparison of mode shapes for Euler-Bernoulli and Timoshenko beam element models. Rotational degrees of freedom, UR3, for vertical modes.

Model A: Element type, beam length, beam height	Model B: Element type; beam length; beam height	MAC-value 1st vertical modes (1.0 = 100% correlation)	MAC-value 2nd vertical modes (1.0 = 100% correlation)
Euler-Bernoulli beam element models:			
B33, 4m, 50mm	B33, 4m, 750mm	1.0	x
Timoshenko beam element models:			
B31, 4m, 50mm	B31, 4m; 200mm	1.0	1.0
B31, 4m, 50mm	B31, 4m; 900mm	1.0	x
B31, 4m, 550mm	B31, 4m; 900mm	1.0	x
Euler-Bernoulli vs Timoshenko beam elements:			
B31, 4m, 50mm	B33, 4m, 50mm	1.0	1.0
B31, 4m, 200mm	B33, 4m, 200mm	1.0	x
B31, 4m, 550mm	B33, 4m, 550mm	1.0	x
B31, 4m, 750mm	B33, 4m, 750mm	1.0	x
B31, 4m; 750mm	B33, 4m, 50	1.0	x

As seen from Table 4.3 modes for different beam heights of Euler-Bernoulli element models correlate 100%, as expected. Since this was the case for a large beam height interval, one test was enough. However, it was unexpected that modes correlate 100% also for all other samples, including modes for the 50 mm high B31-model versus the 900 mm high B31-model. For low slenderness, λ=L/h, larger contributions from shear are in general expected according to common Timoshenko beam theory. The 900 mm high B31-model was thus expected to have some influence from shear compared to the 50 mm high B31-model, since the slenderness of the former is $\lambda = 4.4$, which is a slenderness that supposedly will have shear influence. 100% correlation between a 750 mm high B31-model and the corresponding B33-model, also deviates from expectations, and indicate that there is no influence from shear on Abaqus-calculated mode shapes based on rotational degrees of freedom. Alternatively, it indicates that discovering shear effects by use of MAC is not an adequate method, or simply that MAC analysis does not capture such effects when rotational degrees of freedom mode shapes are compared.

4.5.2 C3D20R – 3D solid element models

Since 3D solid elements have translational degrees of freedom only, only mode shapes represented by vertical and axial degrees of freedom was testable for these models. However, for these elements, which have nodes with corresponding degrees-of-freedom in 3D space, axial deformations in planes away from the neutral axis was possible to investigate. Since the bottom surface displacements – particularly displacements close to the supports – could be influenced by boundary conditions, only degrees-of-freedom at the beam top surface plane was used for mode shape comparisons. The vertical mode shapes of the same 3D solid element models were investigated respectively through the vertical and the axial (horizontal) degrees-of-freedom.

Table 4.4: MAC-values for comparison of mode shapes for 3D solid element models.

Model A: beam length, beam height	Model B: beam length; beam height	MAC-value 1st vertical modes (1.0 = 100% correlation)	MAC-value 2nd vertical modes (1.0 = 100% correlation)
Vertical degrees of freedom, U2, at beam top surface:			
L=5m, h=300mm	L=5m, h=950mm	0.94662	0.57399
L=5m, h=300mm	L=5m; h=500mm	0.99677	0.92016
L=5m, h=300mm	L=5m; h=400mm	0.99944	0.97986
L=5m, h=900mm	L=5m; h=950mm	0.99972	0.99819
Axial degrees off freedom, U1, at beam top surface:			
L=5m, h=300mm	L=5m, h=950mm	0.88964	0.92290
L=5m, h=300mm	L=5m; h=500mm	0.98783	0.96118
L=5m, h=300mm	L=5m; h=400mm	0.99736	0.98363
L=5m, h=900mm	L=5m; h=950mm	0.99966	0.99994

As the results in Table 4.4 show, almost all comparison samples of the first vibration mode are approximately 100% correlated, irrespective of the DOFs for which the mode shapes were calculated. As the respective cases are examples on beam configurations with little difference in slenderness in-between them, this confirms the evident – that mode shapes which should correlate in fact do so.

Only the mode shape comparison of the two configurations in between which the difference in slenderness was relatively the largest, showed noticeable deviations from full correlation. 89 % correlation is just slightly below the usual acknowledged limit for well-correlated modes, although, as mentioned in section 2.4, this limit is not absolute. The observation was expected for the axial mode shape comparison, since the slenderness of the 950 mm beam is much lower than for the 300 mm beam, so eventual shear deformation influence should be noticeably different for the two beam heights. More than 10 % deviations between modes indicates that the MAC-value may have captured this difference, but the statistical significance of a MAC-value is not a question of either/or, and strict conclusions are hard to draw. The deviations may possibly also

have been influenced, or even caused, by other effects such as numerical noise. However, the trend indicating that the larger the difference in slenderness, the larger the difference between mode shapes of two beams, coincides with expectations based on Timoshenko beam theory.

As seen from the results table, one MAC-value for one of the second vertical mode comparisons show particularly low degree of correlation: the vertical degrees-of-freedom mode shape comparison of the 950 mm high beam versus the 300 mm high beam. A 57 % degree of correlation between modes, is far less correlation than for any other samples. Again, a low degree of correlation between these beam configurations fits with expected trends, but in conjunction with the other data for the same configurations, the value seems to be strongly overexaggerated. It is doubtful that it may be evaluated as accurate. If any trend may be read from the data, it is that mode correlation is lesser for the axial degrees of freedom mode shape comparisons than for the vertical degrees of freedom mode comparisons. This points towards a large discrepancy that may have been influenced by other effects such as numerical noise. It could also be the case that the calculation in Abaqus of vertical DOFs for the second mode shape (in the eigenvector) for any of the two beams was numerically inaccurate. For instance, representation of real, physical boundary conditions is not a straightforward process, and such could influence on the eigenvector calculation in the software.

Summarized, evaluating mode shapes for 3D solid element models in DOFs far from the neutral axis seems to fit better with expectations, but the quality on the results are not evident. Since the number of DOFs that make up the mode shapes is larger for the fine meshed 3D solid element models than for the beam elements, according to theory on MAC this should imply that MAC-calculations are more accurate for the 3D solid element models. However, this is no absolute guarantee, and the results should be interpreted with caution.

4.5.3 S8R – Shell element models

Like the beam elements, shell elements have both translational and rotational degrees of freedom. Plane shell elements, like the S8R-element (8 node reduced integrated shell element), only have nodes, and followingly DOFs, in the shell-plane. Orientation of the shell thus influence on whether axial or horizontal displacement degrees-of-freedom mode shapes are relevant to consider.

For the shell models oriented so that the shell thickness is the beam height, all DOFs are localized along the neutral axis horizontal plane, and equivalently as for the beam elements it is neither possible to capture any shear effects through axial deformations for these shell element model configurations. However, rotational degrees-of-freedom about the out-of-plane axis are available for these.

Table 4.5: MAC-values for comparison of mode shapes for shell element models with shell thickness as beam height.

Model A: beam length, beam height	Model B: beam length; beam height	MAC-value 1st vertical modes (1.0 = 100% correlation)	MAC-value 2nd vertical modes (1.0 = 100% correlation)
Vertical DOFs, U3, at beam mid-plane (DOFs perpendicular to shell plane):			
L=4m, h=650mm	L=4m, h=950mm	1.0	1.0
Rotational DOFs, U5, at beam mid-plane:			
L=4m, h=650mm	L=4m, h=950mm	1.0	1.0

As displayed in Table 4.5, resulting MAC-values for this shell orientation show 100% correlation between modes of different beam configurations, for both vertical and rotational degrees-of-freedom mode shapes. This is similar to the results for the beam elements. It may be explained by the shell orientation which make for the node localization to be along the neutral axis plane of the beam only, correspondingly equivalent to the localization of nodes for the beam elements. Anyhow, effects like shear in vertical modes does not seem possible to capture for this element orientation by use of MAC.

For the shell models oriented so that the shell thickness was the beam width, the shell plane was oriented vertically so that both vertical, rotational and axial displacements at the top edge of the beams could be assessed in MAC-analysis.

Table 4.6: MAC-values for shell element models with shell thickness as beam width.

Model A: beam length, beam height	Model B: beam length; beam height	MAC-value 1st vertical modes (1.0 = 100% correlation)	MAC-value 2nd vertical modes (1.0 = 100% correlation)
Vertical DOFs, U2, at beam top surface (shell top edge)			
L=3m, h=350mm	L=3m, h=900mm	0.92589	x
L=3m, h=350mm	L=3m; h=650mm	0.96119	x
Rotational DOFs, U6, at beam top surface (shell top edge):			
L=3m, h=350mm	L=3m, h=900mm	0.96591	x
L=3m, h=350mm	L=3m; h=650mm	0.99027	x
Axial DOFs, U1, at beam top surface (shell top edge):			
L=3m, h=350mm	L=3m, h=900mm	0.93946	x
L=3m, h=350mm	L=3m; h=650mm	0.98949	x

The results displayed in Table 4.6 in overall seem to coincide well with the results for the 3D solid element models. Reduced correlation between modes of beams for which the slenderness is different, fits with the expected. However, for the shell element models, the vertical degrees of freedom mode shapes correlate less than the axial degrees of freedom mode shapes, thus deviating slightly from the 3D solid element

results. Anyhow, these deviations are relatively small, and the observable trends appears to indicate consistency between similar element types.

4.5.4 Summary of MAC-analyses data

For beam elements and shell elements with shell thickness as beam height, MAC calculations displayed 100% correlation between modes in all cases, not only for those where correlation was expected. In a physical sense, the vibration shapes should be similar, and modes were expected to correlate to a high degree, although full correlation was not expected. Usually, 90% correlation is considered the criteria for well correlated modes, but still, deviations from 100% would indicate that mode shapes are not perfectly identical. Since deviations were not present here, it may indicate that shear effects are not large enough to be captured by MAC, or that MAC is not applicable to finding shear effects in mode shapes found from Timoshenko beam element models, or shell models with this shell orientation.

Anyhow, the modal assurance criterion seems to work well in confirming well-correlated modes, which is also one major intention with this tool. And, contrary to the above cases, for 3D solid elements and shell elements, oriented such that the shell thickness represent the beam width (vertically oriented), it seems that effects of shear might still be captured by MAC. At least, modes that are not expected to correlate 100% do not do so, and those that are supposed to correlate do correlate. A particularly good example on the former is the 89 % correlation between the 300 mm and the 950 mm beam modelled with 3D solid elements. According to the 90 % limit for well correlated modes, these beams have modes that do not correlate well, and although it is still a high degree of correlation, it deviates from perfect correlation, and the correlation order of magnitude complies well with expectations related to possible shear influence.

However, it may be that these observations are caused by other effects, such as numerical noise or inaccuracies in some of the degrees of freedom. It may be considered a weakness that MAC does not distinguish potential causes for mismatch in mode correlation. Therefore, it is difficult (if possible) to isolate the influence of shear on a mode shape by use of MAC-calculations. Considering this, MAC seems to work best as a supplement to other methods that are more accurate in determining different parameter contributions and in quantifying effects that may influence on mode shapes and damping.

Other methods could be methods that quantify energy in the system, such as internal energy or other. Although energy parameter investigations could possibly make different element type models more comparable, such investigations have not been performed. The complex frequency step in Abaqus do not calculate any energy parameters.

4.6 Investigating the mass- and stiffness matrices

The Rayleigh damping model has been applied to the models discussed in this chapter. As presented in section 2.2.1 this is a proportional damping model, proportional with the mass matrix and the stiffness matrix, respectively denoted **M** and **K**. The major discrepancies from the analysis in iSight were damping ratios and eigenfrequencies that declined with increasing beam height. For some cases only the damping ratio did decline, while the eigenfrequency continued do increase as expected. As the modes compared seem to be the same, there could be issues with the solving of the problem or the definition of the **K**- and **M**-matrices. Issues with solving of the problem was tested manually in Abaqus and with an alternative method of measuring damping, as discussed in section 2.3.

4.6.1 Mass matrix formulations – consistent versus lumped mass matrix

The mass matrix can be defined by lumping or by consistent shape functions. According to Cook [18], the consistent mass matrix formulation is most accurate for flexural problems, like is the case here, with beams or plates. This higher accuracy is nevertheless negligible if the wavelength of the mode stretches beyond about four elements in the mesh. For this case this means that there is a negligible accuracy to gain from choosing consistent mass matrix formulation over lumped mass matrix. This was tested in Abaqus by running some of the analyses containing discrepancies with lumped mass matrix and consistent mass matrix.

4.6.2 Comparison of analytical and numerical established system mass- and stiffness matrices

The approach to try to figure out causes to the deviations in resulting eigenfrequencies and damping ratios was to investigate the basis for how these parameters are calculated in Abaqus and how the modelling procedure might influence these. The essential is to understand how the system mass- and stiffness matrices are established by the software. Since these matrices are the basis for the calculation of the eigenvectors and eigenvalues, thus influence on the resulting damping matrix, it is necessary to have control over these parameters. However, information in the Abaqus documentation, on how the software establishes the mass- and stiffness matrices, is sparse and non-comprehensive. Particularly, this is the case regarding establishment of the stiffness matrix. Therefore, a test problem was set up analytically for comparison to the Abaqus-established matrices.

The test problem was made as simple as possible, modelling a simply supported beam with two 2-node Timoshenko beam elements. Thus, shear deformations were accounted for in establishment of the stiffness matrix.

4.6.2.1 Formulation of the analytical model

For the analytical model the element mass and stiffness matrices were established on consistent format, for which the element mass and stiffness relations were based on a two-node beam element with four DOFs – one vertical and one rotational degree-of-freedom in each node. The system thus had three global nodes, in which four user-defined kinematic DOFs, $\boldsymbol{r} = [r_1\ r_2\ r_3\ r_4]^T$, were introduced for the global system: r_1 – rotational degree of freedom at left support, r_2 – vertical degree of freedom at mid-span, r_3 – rotational degree of freedom at mid-span, and r_4 – rotational degree of freedom in right support.

Further, the element compatibility matrices, **a**, were established and used to formally establish the system stiffness matrix by the direct stiffness method: $\mathbf{K} = \sum_{\mathbf{i=1}}^{\mathbf{\#el}} (\mathbf{a^i})^{\mathbf{T}} \mathbf{k^i a^i}$ **.** The system mass matrix, **M**, was established equivalently: $\mathbf{M} = \sum_{\mathbf{i=1}}^{\mathbf{\#el}} (\mathbf{a^i})^{\mathbf{T}} \mathbf{m^i a^i}$.

Beam configuration parameters for the analytical model were 6m beam length (L = 6m), 0.1m beam width (w = 0.1m) and 0.2m beam height (h = 0.2m). Other necessary geometric input parameters calculated were the cross sectional area, A, the second moment of inertia, I, about the strong axis, the shear area, A_s and the shear coefficient, α_s that introduce shear stiffness into the element stiffness relation. The shear factor, $\kappa = \frac{6}{5}$, for rectangular cross sections, was used for determining the shear area used for calculating α_s.

Specifying material parameters was done respectively according to an isotropic and an orthotropic definition. Input material strength parameters in both cases were the elastic modulus, E_1, along the grain direction and the (transverse) shear modulus, G_{12}. For the isotropic definition, G is calculated through the relation $G = \frac{E}{2(1+\nu)}$. ν is theoretically bounded to be below 0.5 [19]. It follows from the relation that the shear stiffness will be too large for timber, even for $\nu = 0.5$. Thus, an isotropic model fails to describe timber for all values of ν, but to carry out the comparison ν was set to 0.4, since the Poisson's ratio is limited to be below 0.5 for an isotropic model in Abaqus. For the orthotropic material variant of the analytical model, the three shear stiffnesses are defined as previously in this work. For convenience, these are repeated in Table 4.7.

For both the analytical and the numerical model material density and strength parameters (except $G_{isotrop}$, and rolling shear G_{23}) were used as specified for timber quality C24 in NS-EN338:2016 [14] presented in Table 4.7. Rayleigh damping coefficients were set to α_R= 1.13 and β_R = 7.47×10^{-5}. Due to Abaqus unit conventions, input units differed between the analytical and numerical models. For convenience, input values with respective units for both analytical and numerical models are presented in Table 4.7.

Table 4.7: Input values in numerical models for matrix investigations.

Analytical models			
E = 11 000 MN/m²	$G_{isotrop}$ = 3928.6 MN/m²	ν = 0.4	Density, ρ = 420kg/m³
	$G_{orthotrop}$ = 690 MN/m²		
Abaqus models, isotropic material definition			
E = 11 000 N/mm²		ν = 0.4	Density, $\rho = 4.2\times10^{-10}$ ton/mm³
Abaqus models, orthotropic material definition			
E_1 = 11 000 N/mm²	G_{12} = 690 N/mm²	ν_{12} = 0.5	Density, $\rho = 4.2\times10^{-10}$ ton/mm³
E_2 = 800 N/mm²	G_{13} = 690 N/mm²	ν_{13} = 0.6	
E_3 = 400 N/mm²	G_{23} = 30 N/mm²	ν_{23} = 0.6	

Eigenfrequencies were calculated by use of the "eig"-function in MATLAB for both the analytical models and the numerical models. All calculated eigenfrequencies in this section are thus undamped frequencies. However, as discussed in section 3.4.2, for low levels of damping the difference between undamped and damped eigenfrequencies should be negligible according to theory. This was also confirmed to be the case throughout investigations of the frequency step and the complex frequency step in the Abaqus models.

Units for the entries in the consistently formulated stiffness- and mass matrices are respectively:

$$[\mathbf{K}] = \begin{bmatrix} \text{Nm} & \text{N} & \text{Nm} & 0 \\ \text{N} & \frac{\text{N}}{\text{m}} & 0 & \text{N} \\ \text{Nm} & 0 & \text{Nm} & \text{Nm} \\ 0 & \text{N} & \text{Nm} & \text{Nm} \end{bmatrix}, [\mathbf{M}] = \begin{bmatrix} \text{kg m}^2 & \text{kg m} & \text{kg m}^2 & 0 \\ \text{kg m} & \text{kg} & 0 & \text{kg m} \\ \text{kg m}^2 & 0 & \text{kg m}^2 & \text{kg m}^2 \\ 0 & \text{kg m} & \text{kg m}^2 & \text{kg m}^2 \end{bmatrix}, [\text{f}_\text{n}] = \text{Hz}.$$

Analytically established consistent mass matrix, $\mathbf{M^a_{cM}}$, stiffness matrix, $\mathbf{K^a_{iso}}$, and eigenfrequencies $\mathbf{f^a_{n,iso}}$ for isotropic material definition become:

$$\mathbf{K^a_{iso}} = \begin{bmatrix} 966987.87 & -481695.62 & 478098.99 & 0 \\ & 642260.83 & 0 & 481695.62 \\ & Sym. & 1933975.75 & 478098.99 \\ & & & 966987.87 \end{bmatrix},$$

$$\mathbf{M^a_{cM}} = \begin{bmatrix} 2.16 & 2.34 & -1.62 & 0 \\ & 18.72 & 0 & -2.34 \\ & Sym. & 4.32 & -1.62 \\ & & & 2.16 \end{bmatrix}, \text{and}$$

$$\mathbf{f^a_{n,iso,cM}} = \begin{bmatrix} 12.93 \\ 57.24 \\ 143.00 \\ 260.36 \end{bmatrix}, \text{where superscript "a" refers to "analytical model".}$$

For the analytical model with shear modulus defined according to the orthotropic material definition, the mass matrix remains unchanged, while the stiffness matrix is changed and eigenfrequencies become slightly changed, giving

$$\mathbf{K}^{\mathbf{a}}_{\mathbf{ortho}} = \begin{bmatrix} 920312.67 & -450578.81 & 431423.77 & 0 \\ & 600771.74 & 0 & 450578.81 \\ & Sym. & 1840625.31 & 431423.77 \\ & & & 920312.65 \end{bmatrix}, \text{ and}$$

$$\mathbf{f}^{\mathbf{a}}_{\mathbf{n,ortho,cM}} = \begin{bmatrix} 12.84 \\ 57.24 \\ 139.19 \\ 251.81 \end{bmatrix}.$$

Establishment of analytical system matrices (for shear-flexible (Timoshenko) element option) for both isotropic and orthotropic material definition (shear stiffness only changed parameter) and calculation of frequencies from solving the EVP is performed in MATLAB – a script is provided in Appendix A.

In comparison, the eigenfrequencies calculated with the analytical formula given in Equation (3-8) are $\mathbf{f}^{\mathbf{formula}}_{\mathbf{n}} = [12.89\ 51.57\ 116.03\ 206.28]$. For the record, it should be repeated that these are based on Euler-Bernoulli beam theory, and they are therefore not directly comparable to the frequencies calculated for models consisting of Timoshenko elements as is the case here – both eigenfrequencies from the analytically established EVP and from the Abaqus models are expected to be relatively lower, since shear deformations are included in these. Thus, the eigenfrequencies calculated from the EB-based formula – hereafter referred to as the analytical EB eigenfrequencies – are only included here as a reference for comparison.

As seen, the change from isotropic to orthotropic material definition has minor influence on the eigenfrequencies; particularly the two first eigenfrequencies, which are the ones of interest here, seem practically unaffected – the relative change between the first natural frequencies is -0.7%, and no difference between the second natural frequencies. Compared to the analytical EB eigenfrequencies, Abaqus overestimates all frequencies above the first, thus indicating a stiffer behaviour in the numerical model the higher the vibration mode. This is not surprising, since the analytical model consist of only two 2-node-elements and thus will not be able to accurately represent modes other than the first. This is also the case for the numerical model. Comparing frequencies other than the first natural frequency is therefore not of relevance in any physical sense. The behaviour decreases for increasing number of elements – with sufficiently many elements it should not be an issue for the modes of interest.

4.6.2.2 Output matrices from Abaqus test models

The numerical Abaqus models were modelled by two B21-Timoshenko beam elements, and material properties was modelled according to Table 4.7. The frequency extraction

step was executed prior to the complex frequency step used for calculation of damping. The SIM architecture was used, and projection of damping operators was toggled on in the frequency step definition.

Several options are possible for the format in which Abaqus writes the system matrices, however, all are on sparse format, i.e. they are a list of all nonzero contributions to the system matrix. Matrices must therefore be established on full format in order to be readable.

Abaqus does not allow for introduction of user-defined kinematic degrees-of-freedom, and consequently writes the system matrices with contributions for all DOFs. Since the B21-element in Abaqus has two nodes and three DOFs in each node (two translational and one rotational degree-of-freedom), for a system consisting of three nodes this means that the global system has a total of 3x3 = 9 DOFs. It follows that the system matrices become 9x9. In order to compare the Abaqus matrices to the analytically derived matrices, columns and rows has to be deleted in the Abaqus output-matrices that corresponds to DOFs other than the kinematic, i.e. the constrained DOFs at supports, as well as axial degrees-of-freedom along the beam length. An example MATLAB script for extraction and transcription of system matrices from Abaqus, as well as calculation of frequencies based on those, is provided in Appendix D.

Due to Abaqus unit conventions (Abaqus is unitless) and choice of units in the modelling procedure, the units for matrix entries in the output Abaqus matrices are *tons* for weight and *mm* for distance. For comparison purposes, the Abaqus output matrices presented in the following are scaled according to units of *kg* for weight and *m* for distance.

Two changes were made in the numerical models from the starting point, which was isotropic material definition and lumped mass matrix, which is the default setting for the mass matrix. Changing material definition to orthotropic and mass matrix format to consistent was first done separately in order to evaluate the influence from each change on the eigenfrequencies. The analytical and numerical models with isotropic material definition and consistent mass matrix could then also be compared. Both changes were introduced in the same model for comparison to the corresponding analytical model. The different numerical models will be referred to chronologically as cases 1 to 4 below. In the matrix notation, superscripts “A” refers to Abaqus models and “a” to analytical models.

The analytical matrices are repeated here for convenience:

$$\mathbf{M}_{\mathbf{cM}}^{\mathbf{a}} = \begin{bmatrix} 2.16 & 2.34 & -1.62 & 0 \\ & 18.72 & 0 & -2.34 \\ & Sym. & 4.32 & -1.62 \\ & & & 2.16 \end{bmatrix}$$

$$\mathbf{K}_{\mathbf{iso}}^{\mathbf{a}} = \begin{bmatrix} 966987.87 & -481695.62 & 478098.99 & 0 \\ & 642260.83 & 0 & 481695.62 \\ & Sym. & 1933975.75 & 478098.99 \\ & & & 966987.87 \end{bmatrix}$$

$$\mathbf{K}_{\mathbf{ortho}}^{\mathbf{a}} = \begin{bmatrix} 920312.67 & -481695.62 & 431423.77 & 0 \\ & 600771.74 & 0 & 450578.81 \\ & Sym. & 1840625.31 & 431423.77 \\ & & & 920312.65 \end{bmatrix}$$

For the initial model (case 1) with isotropic material definition and lumped mass matrix, the resulting stiffness matrix, $\mathbf{K}_{\mathbf{iso}}^{\mathbf{A,1}}$, and first natural eigenfrequency, $f_{n,1,iso}$ become:

$$\mathbf{K}_{\mathbf{iso}}^{\mathbf{A,1}} = \begin{bmatrix} 1119366.47 & -583281.35 & 630477.58 & 0 \\ & 777710.00 & 0 & 583281.35 \\ & Sym. & 2238732.93 & 630477.58 \\ & & & 1119366.47 \end{bmatrix}$$

and $f_{n,1,iso}^{A,1} = 13.06$ Hz (all higher frequencies are strongly over estimated).

Apparently, the values of all stiffness matrix entries are comparatively higher than for the corresponding analytical model. This also results in the first natural frequency being slightly over estimated, although the relative change between $f_{n,1,iso}^{A,1} = 13.06$Hz and $f_{n,1,iso}^{a} = 12.93$Hz (analytical model) is just 1%.

Changing the mass matrix format to consistent (case 2) leaves the stiffness matrix unchanged, and changes respectively the mass matrix, $\mathbf{M}_{\mathbf{iso,cM}}^{\mathbf{A,2}}$, and eigenfrequencies, $\mathbf{f}_{\mathbf{n,iso}}^{\mathbf{A,2}}$, into:

$$\mathbf{M}_{\mathbf{iso,cM}}^{\mathbf{A,2}} = \begin{bmatrix} 2.16 & 2.35 & -1.63 & 0 \\ & 20 & 0 & -2.35 \\ & Sym. & 4.32 & -1.62 \\ & & & 2.16 \end{bmatrix}, \text{ and } \mathbf{f}_{\mathbf{n,iso}}^{\mathbf{A,2}} = \begin{bmatrix} 13.13 \\ 57.13 \\ \gg 100 \\ \gg 100 \end{bmatrix}.$$

Observing the mass matrix, $\mathbf{M}_{\mathbf{iso,cM}}^{\mathbf{A,2}}$, it is approximately identical to the consistently established analytical mass matrix, $\mathbf{M}_{\mathbf{cM}}^{\mathbf{a}}$. The relative change, when going from lumped to consistent mass matrix, between frequencies $f_{n,1,iso}^{A,1} = 13.06$ Hz and $f_{n,1,iso}^{A,2} = 13.13$Hz is only 0.5%, indicating little influence in changing the mass matrix format. Although this difference is small, it could be noted that lumped formatted mass matrix apparently gives a first natural frequency closer to the analytical.

Keeping the mass matrix lumped and changing material definition to orthotropic (case 3) results in the stiffness matrix, $\mathbf{K}_{\mathbf{ortho}}^{\mathbf{A,3}}$, and eigenfrequencies, $\mathbf{f}_{\mathbf{n,ortho}}^{\mathbf{A,3}}$:

$$\mathbf{K}_{\mathbf{ortho}}^{\mathbf{A,3}} = \begin{bmatrix} 398112.57 & -102445.41 & -90776.32 & 0 \\ & 136590.00 & 0 & 102445.41 \\ & Sym. & 796225.13 & -90776.32 \\ & & & 398112.57 \end{bmatrix},$$

$$\mathbf{f}_{\mathbf{n,ortho}}^{\mathbf{A,3}} = \begin{bmatrix} 9.18 \\ \gg 100 \\ \gg 100 \\ \gg\ 100 \end{bmatrix}.$$

Orthotropy was modelled in the material property module by specifying the engineering constants presented in Table 4.7.

Changing to orthotropic material definition obviously influence greatly on the stiffness matrix and followingly on the frequencies and damping ratio. The relative change in the first natural frequency between numerical models (mass matrix lumped) with isotropic ($f_{n,1,iso}^{A,1} = 13.06$ Hz) and orthotropic ($f_{n,1,ortho}^{A,3} = 9.18$Hz) material definitions, is as high as -29.7%. This is quite inaccurate compared to the analytical frequencies calculated from the EVP with orthotropic material definition – also compared to the analytic EB eigenfrequencies, and although they should be lower than these, they are much lower than expected.

These observations remain rather unchanged for the last investigated case (case 4) in which both the mass matrix is formulated on consistent format and the material definition is orthotropic. Since changing the material definition to orthotropic influence the stiffness matrix only (since it does not include a change in material density) and changing the mass matrix format changes the mass matrix only, when changing both in the same model, both matrices are not different relative to the respective corresponding cases above. Thus, the stiffness matrix is

$$\mathbf{K}_{\mathbf{ortho}}^{\mathbf{A,4}} = \begin{bmatrix} 398112.57 & -102445.41 & -90776.32 & 0 \\ & 136590.00 & 0 & 102445.41 \\ & Sym. & 796225.13 & -90776.32 \\ & & & 398112.57 \end{bmatrix},$$

and mass matrix $\mathbf{M}_{\mathbf{cM}}^{\mathbf{A,4}} = \begin{bmatrix} 2.16171 & 2.35 & -1.63229 & 0 \\ & 20 & 0 & -2.35 \\ & Sym. & 4.32342 & -1.62 \\ & & & 2.16 \end{bmatrix}$.

The resulting eigenfrequencies now become $\mathbf{f}_{\mathbf{n,ortho,cM}}^{\mathbf{A,4}} = \begin{bmatrix} 9.92 \\ 57.13 \\ \text{nr} \\ \text{nr} \end{bmatrix}$.

Although in this case, changing to orthotropic material definition still gives a first eigenfrequency seemingly too low, the difference is somewhat smaller compared to case 3 above. However the difference is still evident: the relative change in first natural frequency between analytical and numerical model, for which the frequencies are respectively $f_{n,1,ortho,cM}^{a} = 12.84$Hz and $f_{n,ortho,cM}^{A,4} = 9.92$Hz, is -22.7%. This is a relatively

large discrepancy and should not be ignored. The deviations between analytically and numerically established stiffness matrices are also still evident.

However, although the discrepancy is large and could be one cause to the observed deviations in results in chapter 4, another factor, that always influences on results, is the number of elements in the model. In general, for a beam model consisting of two elements only, results are not likely to be very accurate. As already mentioned, this is particularly the case with respect to calculating the vibration mode shapes.

The influence from number of elements was tested by increasing the number of elements in the latter numerical model to five and 10. With five elements the resulting first eigenfrequency is 12.09 Hz which is closer to the analytical calculated from solving the EVP, and also more realistic compared to the analytical EB eigenfrequencies. Increasing to 10 elements the frequency become 12.59 Hz, which is even closer. Using 20 elements results in a first eigenfrequency of 12.73 Hz, meaning that the relative change between analytical and numerical frequency is less than one percent.

Initial tests during the early-phase work of this study also revealed that the accuracy gain in going from 20 elements to more than 20 was in an order of magnitude less than 1%, no matter how many elements. This was also the argument for the choice of using minimum 20 elements in each model investigated. Most of the models had the first eigenfrequencies below the upper limit of 100Hz. These models had 40 elements or more, thus, number of elements should not be the governing cause for discrepancies.

Since the Abaqus mass matrix is approximately equivalent to the analytical mass matrix, the issue should neither be related to establishment of this. The resulting stiffness matrices from Abaqus are, however, not identical to the analytical stiffness matrices as they are established here. Thus, Abaqus establishes the stiffness matrices differently. Indications are thus strong that the deviating results discussed in section 4.3 is related to how the system stiffness matrix is established, since Rayleigh damping is proportional to both $\mathbf{K}$ and $\mathbf{M}$.

As mentioned, information in the documentation is sparse considering this topic. This could be due to commercial reasons. Since Abaqus is a commercial software, it could be the case that some details are intentionally left out of the documentation for competitive advantage.

It should be noted that the Timoshenko element is formulated to be a general-purpose element, an element that is efficient for both slender beams (EB theory) and thick beams (Timoshenko theory). As a result, the formulation might be quite complex and this could also be the cause for discrepancies in damping ratio and the reason to why the analytical stiffness matrix does not match with the outputted one from Abaqus [7].

5 Other methods for applying damping

In this work the main method for applying damping has been Rayleigh damping at the material level. As discussed in section 2.2, damping can be characterized differently regarding how it is traced back to its origin. Due to this it is probably necessary to combine different mathematical models to simulate damping better. This has been tested to some extent in this work.

5.1 Global Rayleigh damping

Abaqus offers an opportunity to apply global viscous damping as Rayleigh damping, in addition to the material level Rayleigh damping. This cannot be applied through the Abaqus interface directly but must be specified in an input file [8]. An example of this is found in Figure 5.1. The line outlined in the red box must be added in the section of the input-file where the complex frequency step is defined, as shown in the extract in the figure.

To run an analysis from an input-file one first need the input-file. This can be obtained by creating a job and instead of submitting it choose "Write input" – then an input-file corresponding to the job is created in the work directory. This input-file must then be edited, and the part outlined in Figure 5.1 is added. Finally, to analyse this in Abaqus a new job is created, and in the pop-up window the source is changed from the default "Model" to "Input file".

```
** STEP: ComplFreq
**
*Step, name=ComplFreq, nlgeom=NO, perturbation, unsymm=YES
*Complex Frequency, friction damping=NO
*GLOBAL DAMPING, ALPHA=1.13, BETA=7.47E-05
, , ,
**
** OUTPUT REQUESTS
**
**
** FIELD OUTPUT: F-Output-2
**
*Output, field, variable=PRESELECT
*End Step
```

Figure 5.1: Extract from a .inp-file from Abaqus, also referred to as input-file. The excerpt shows how to apply global damping to the complex frequency step.

This global damping is supported in modal analyses, direct-solution steady-state dynamic analysis, complex eigenvalue extraction, matrix or substructure generation, and will be ignored for other analyses.

To see how the global damping is applied, a series of beam configurations, like the ones in section 4.3, was ran. As seen earlier, the beams spanning one and two meters contribute with few data points, so these was left out. The beams analysed were Timoshenko beams (B31 elements) and all were run once with SIM architecture and once without (traditional architecture).

The results for the damping ratios are plotted for the eigenfrequency of the corresponding mode; this is shown in Figure 5.2 and Figure 5.3. The curve in the plot is the same as the "Rayleigh-curve" described in section 4.3.4. Figure 5.2 displays results with traditional architecture (not SIM) and one can see that all the damping ratios match the Rayleigh-curve for their given eigenfrequency. It was also evident from the results that the eigenfrequencies increased as expected for taller beams. This can be seen from the figure as there is no clustering of data points from the same beam length for any frequencies.

Figure 5.3 show the same as Figure 5.2, except that SIM architecture is applied. Now the damping ratios display a completely linear relation to the eigenfrequencies, and by that, do not match the Rayleigh-curve. This was not expected.

As explained in section 2.2.1 the Rayleigh damping model is a proportional damping model and it can be seen from the expression that the β-term gives a contribution to the damping matrix that is linearly dependent on frequency. This is also shown in Figure 5.3. To check for this a new analysis was run with the original α-coefficient included, but the β-coefficient was set to zero. Results showed zero damping for all modes. The opposite test was also run with α set to zero and β set to its original value and the results were identical to what is presented in Figure 5.3.

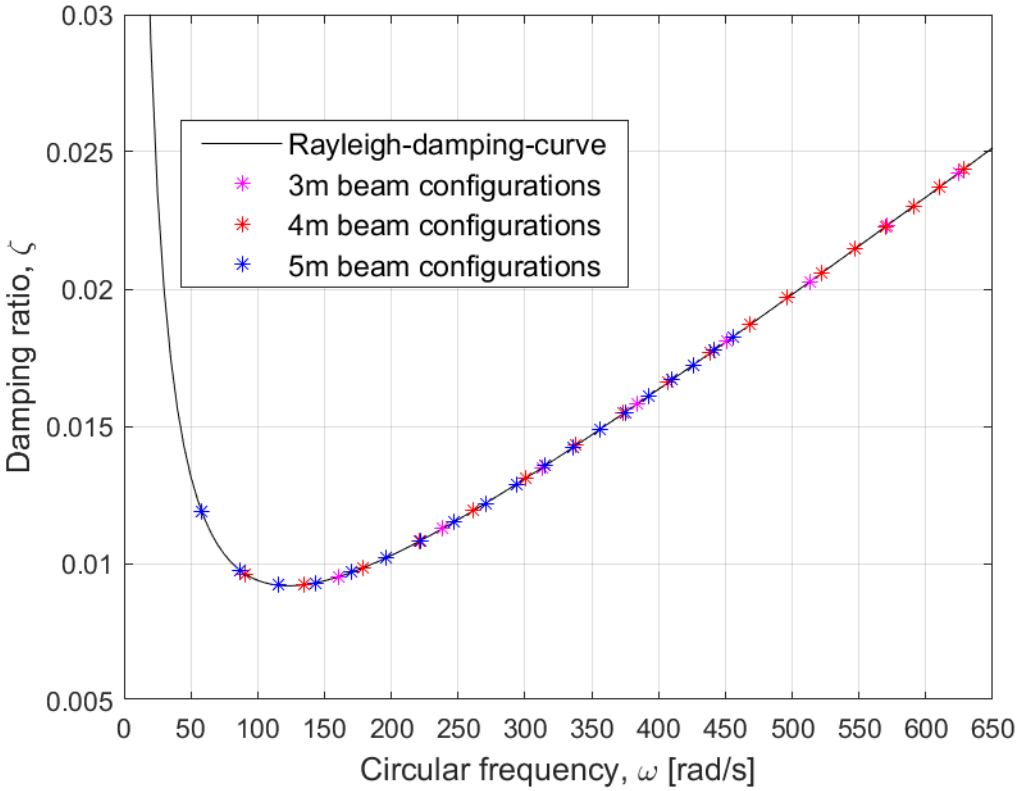

Figure 5.2: Timoshenko beam model. Global damping applied and SIM turned off. Damping ratios plotted against the eigenfrequency, with the Rayleigh-curve. First vertical mode.

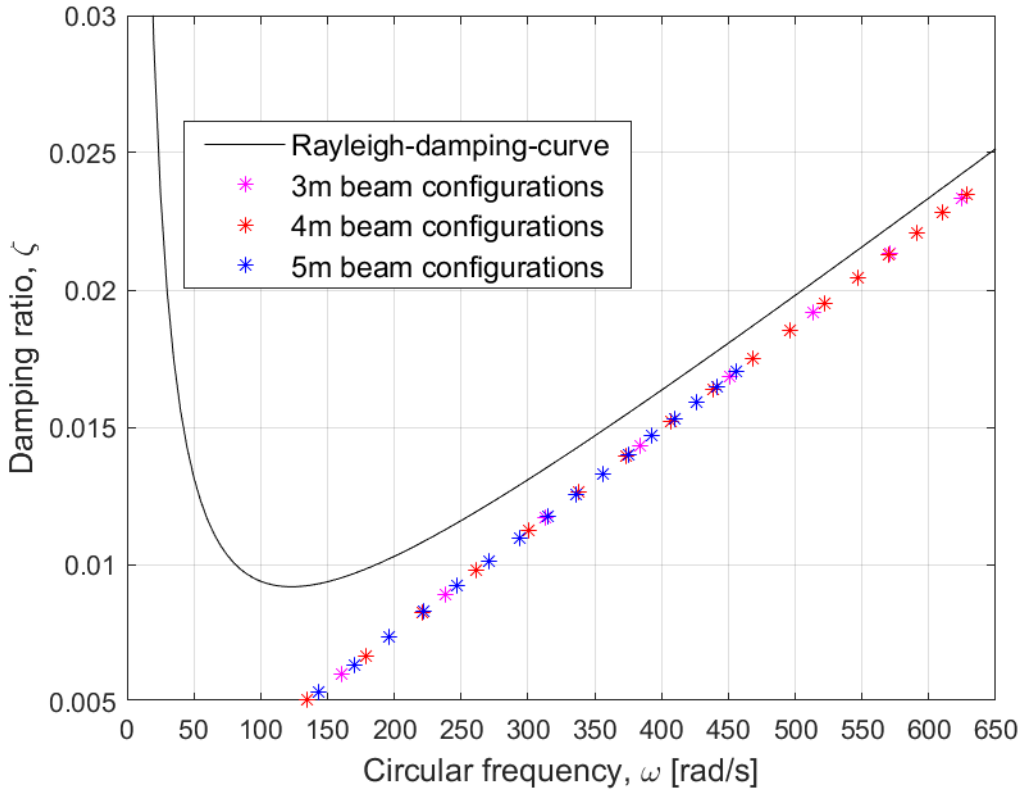

Figure 5.3: Timoshenko beam model. Global damping and SIM applied. Damping ratios plotted against the eigenfrequency, with the Rayleigh-curve. First vertical mode.

More analyses were run to inspect closer how the eigenfrequencies and the damping ratios from a global application of damping compare to a material/element application. This was tested for two simply supported beams with different dimensions, mesh and elements. Table 5.1 describe the modelled beams.

Table 5.1: Description of the beams tested for global and material damping. L = span length, w = beam width and h = beam height.

	Beam elements	No. of elements	Dimensions (L x w x h) [mm]
2D beam	B21 (Timoshenko)	2	6000x100x200
3D beam	B31 (Timoshenko)	100	5000x100x400

For both beams the same Rayleigh coefficients were added at a global level and at the material level. For the material level the same material was used for the whole beam. This was done in three different analyses:

- Only material level
- Only global level
- Both material and global level

Table 5.2: Damping ratios for different ways of applying the same Rayleigh damping coefficients.

	Damping ratio		
	Material level	Global level	Both levels
2D beam	0.928 %	0.995 %	1.92 %
3D beam	1.01%	0.829 %	1.84 %

Results show that for the 2D beam the damping is higher when applied at a global level than at the material level, but for the 3D beam it is opposite. The damping level is about 1 %. The difference between adding in the material level and the global level, when measured in percent, is almost 20 % for the biggest deviation. For both the analyses the case with application at both levels are, for three significant figures, the same as summing the results from the two separate analyses.

The results presented here indicate that with global Rayleigh damping and SIM architecture, Abaqus will in this case ignore the α-coefficient and include only the linear part (the β-part) of the Rayleigh-model. An answer to why these unexpected results occurred with the SIM architecture has not been found, but it raises suspicion regarding the reliability of the damping model in Abaqus.

For the comparison of the material and global level, one could assume that the damping would be the same when the material is the same throughout the whole model. For a simple two element beam presented in Appendix G, this has been shown to be analytically true. The authors have not succeeded in finding out why this is not the case in Abaqus.

5.2 Composite modal damping

One of the choices for damping when configuring the material in Abaqus is composite modal damping. This type of damping could be beneficial since it is applied at the material level and one can then differentiate damping between components in a structure.

The composite modal damping has some limitations. It can only be applied to modal-based procedures based on eigenvalue extraction with the Lanczos eigensolver and not SIM architecture, or eigenvalue extraction with subspace iteration. The exception is steady state dynamics with subspace projection [20]. This leaves composite modal damping not applicable for complex frequency step or time integration, the two main methods for measuring damping in this work.

The analysis step “steady-state dynamics, modal” can include composite modal damping, but the Rayleigh damping from the material section is ignored. Rayleigh damping has been the main method for applying damping and as these two seem to exclude one another the composite modal damping was not examined any further.

6 Assembled elements

6.1 Combining different levels of damping

When assembling different parts into a complete structure, one rarely uses only one type of part. Different parts are likely to have different levels of damping; therefore, it is useful to have knowledge on how different levels of damping combined interact in an assembled component or a complete structure.

A set of simple analyses was carried out as a start. A beam was modelled as 3D solid, spanning 6 m, simply supported, with cross section of 100x200 mm, put edgewise and restricted to only vertical modes. This beam was made in three different versions. The first was split horizontally in two halves (Figure 6.1(a)) , number two was split vertically in two halves (Figure 6.1(b)) and number three was split vertically in three pieces with the middle part holding 50 % of the beam and the two other pieces 25 % symmetrically on each side (Figure 6.1(c)). This was done to apply different materials to the different parts of the beam. The splitting was made by partitioning in Abaqus and the rest of the modelling follow the guide in section 4.2.

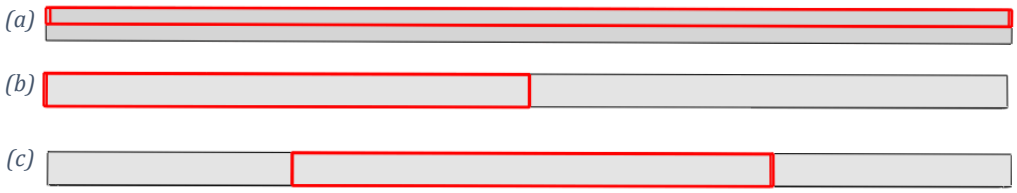

Figure 6.1,a-c: The different versions of the beam made for combining different levels of damping. The highlighted part is the part that the material is changed in for the different runs.

Three different materials were configured, with the only difference between them being the level of Rayleigh damping assigned. The Rayleigh-coefficients are presented in Table 6.1, respectively for material 1, material 2 and material 3. As seen, the level of damping is doubled from one material to the next.

Tabell 6.1: Rayleigh coefficients for the different materials that were combined.

	Material 1	Material 2	Material 3
Rayleigh-coefficient, α_R	1.13	2.26	4.52
Rayleigh-coefficient, β_R	7.47×10^{-5}	1.49×10^{-4}	2.98×10^{-4}

Every version of the beam was analysed in three runs, every run with a different set of materials. The first run was with only material 1 in the whole beam, the second with material 1 and material 2, and the third with material 1 and material 3. Which parts that held which material for the different runs depended on how the beam was split for the relevant version of the beam.

For the horizontally split beam the top material was changed for each respective run. In the version split vertically in the middle it was the front part that was changed. For the last version the middle 50 % of the beam was changed.

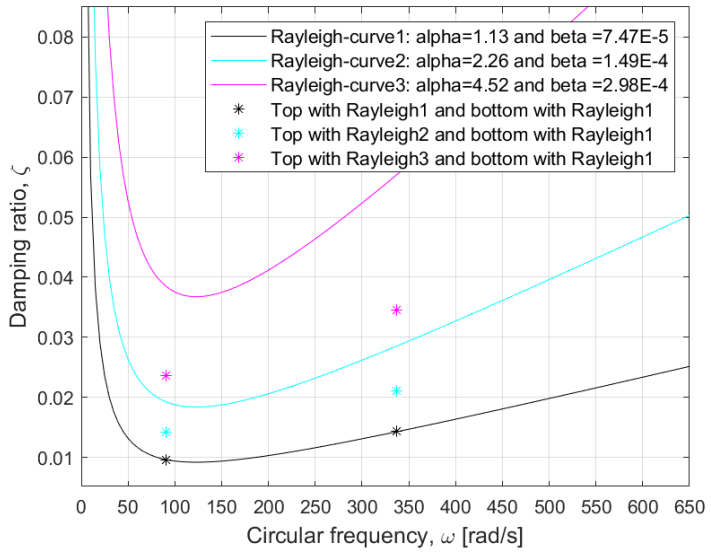

Figure 6.2: Beam split horizontally. Resulting damping ratio for the first two vertical modes are shown as stars. The curves represent the mathematical expression of three different Rayleigh damping levels.

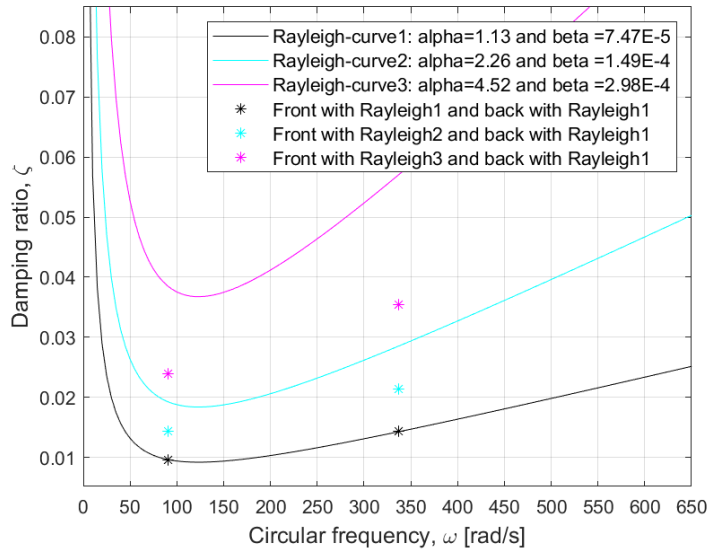

Figure 6.3:Beam split vertically once. Resulting damping ratio for the first two vertical modes are shown as stars. The curves represent the mathematical expression of three different Rayleigh damping levels.

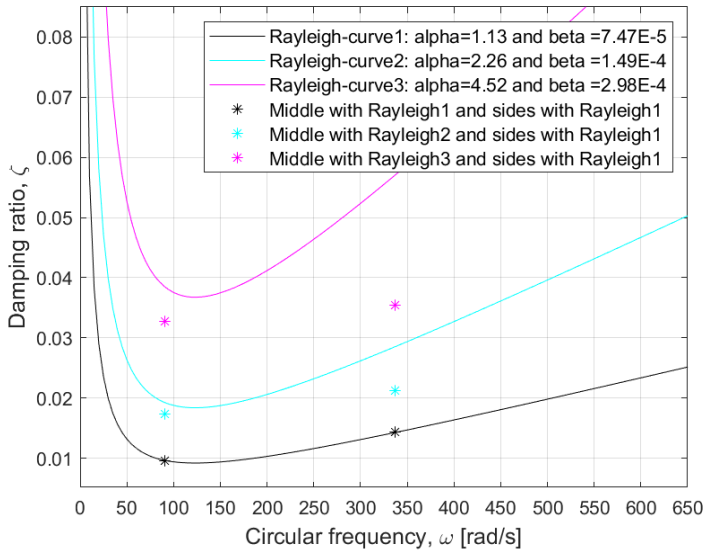

Figure 6.4: Beam split vertically twice. Resulting damping ratio for the first two vertical modes are shown as stars. The curves represent the mathematical expression of three different Rayleigh damping levels.

Figure 6.2, Figure 6.3 and Figure 6.4 show the results from the analysis. In each figure, the differently coloured data points represent the results from the different runs. The three leftmost single data points correspond to the first vertical mode and the three to the right correspond to the second vertical mode.

A first observation is that the run with the beam with only one material (black coloured data points) display damping that match the relevant Rayleigh curve exactly. These data points are present and identical in all the three figures. For the runs with combinations of materials, several observations can be made. For the first vertical mode one can see that the damping ratio is somewhere between the Rayleigh curves of the materials implemented. For the horizontally split and the vertically split in two parts the results are approximately equal and are in the middle of the two Rayleigh curves for the respective materials. The single value point for the beam vertically split in three parts is, on the other hand, located much closer to the curve for the material in the midsection. The second vertical mode has less difference between the different runs. The two vertically split versions have almost equal damping, and the horizontally split has just a little lower damping than the other two, with a difference of about +/- 0.001 for the damping ratio.

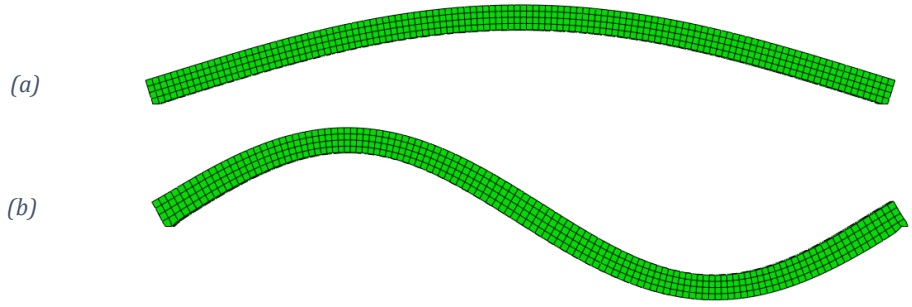

Figure 6.5: First vertical mode (a) and the second vertical mode (b) for the beam.

To suggest an explanation for these results, the two first vertical modes are provided as an illustration in Figure 6.5. Recalling the results for the second mode, the damping was quite similar regardless of how the material was distributed. For the three different versions of the beam, the two symmetric bays of the second mode hold equal amounts of material, and how much the different materials will be displaced is quite similar. For all three versions there will be equal amounts of material in the maximum displaced regions and in the regions around the three nodes that has zero displacement.

When regarding the same, i.e. how much material that is present in the parts with different amount of displacement, the case is similar for two out of three beam versions, in the first vertical mode. These two are the horizontal split and the vertical split in two pieces. On the other hand, for the beam version split vertically in three pieces, the midsection of the beam - which contain the higher damped material - covers the part of the beam where mode shape displacements are largest. The mode shape displacements are much lower towards the beam ends that contain the lower damped material. This version for the first mode is clearly the one that deviates the most from the rest. These observations suggest that the material in the part with the greatest displacements in the vibration mode, will have the greatest influence on the resulting damping.

6.2 Damping in 2D frame models

To move further towards modelling of a complete structure, 2D frames were modelled. These frames were chosen to be restricted to 2D to increase the complexity of the model with a small enough incremental change for the results to be assessed in a good way. Additionally, to limit the complexity, all the joints were modelled as fixed, the floor heights were the same for all versions, and the beam lengths were either 6 meter for the whole model or 12 meter for the whole model.

The models created is described by the following:

- Column cross section: 500x500 mm
- Beam cross section: 495x140 mm
- Material: C24, as presented in Table 4.1
- Floor height (system line): 3 m
- Beam span (between column system lines): 6 m or 12 m
- Element type: Timoshenko 2D beam (B21) or 8-node plane stress (CPS8R)
- Maximum 5 floors and maximum 5 spans

6.2.1 Initial tests of damping type combinations

Initially with the 2D frames, tests with material damping, global damping and SIM architecture was run. These were done on eight-node plane stress elements (CPS8R). These tests showed differing results with the SIM architecture and with the traditional

architecture (SIM turned off). With SIM architecture turned off, the structural material damping is ignored, and material and global Rayleigh damping yield identical results when added separately.

SIM turned on will include structural damping and Rayleigh damping added as material damping, as well as global Rayleigh damping. The structural damping and the Rayleigh damping seem to work fine as material damping, but the global Rayleigh damping shows unexpected results. The first mode of the frame was considered. This showed a negative damping ratio for a frequency about 10 Hz, and the three following modes showed positive damping ratios.

This was investigated further by adding only the α_R-part of the Rayleigh damping, only the β_R-part, and the same α_R-part with negative sign and the normal β_R-part. With only α_R, all the modes yield negative damping, only β_R yield positive results as expected, and with negative α_R and positive β_R all the modes have positive damping. The results for the last case are not the same as global Rayleigh damping without SIM, but close. The same frame was modelled with Timoshenko beam elements and the results with positive and negative damping ratios did not occur for that model when the same tests was run.

Because of these results all the frames were modelled with traditional architecture, one set with material Rayleigh damping and one set with global Rayleigh damping. The results from these analyses are presented in the following. The frame configurations are named “Configuration #Floors#Spans”, i.e. “#Floors” correspond to the first integer in the names, “#Spans” to the second integer.

In the following, results are only included for the frames with beam spans of 6m. Results for the frames with 12m beam spans were similar.

6.2.2 2D frames – 8-node continuum plane stress elements (CPS8R) – material/global Rayleigh damping

Figure 6.6 and Figure 6.7 display the damping ratio, plotted towards the Rayleigh curve, for all calculated modes for all modelled frame configurations with section spans of 6 meter, and modelled with 8-node plane stress elements (CPS8R). The two figures represent damping modelled as material Rayleigh damping or as global Rayleigh damping – the results were identical between the two damping types, giving the same eigenfrequencies and damping ratios. As the figures display, the damping ratios coincide with the Rayleigh curve for all modes for all frame configurations, irrespective of frame size (meaning number of floors and sections). In general, the highest number of eigenfrequencies are found for higher frames with more floors.

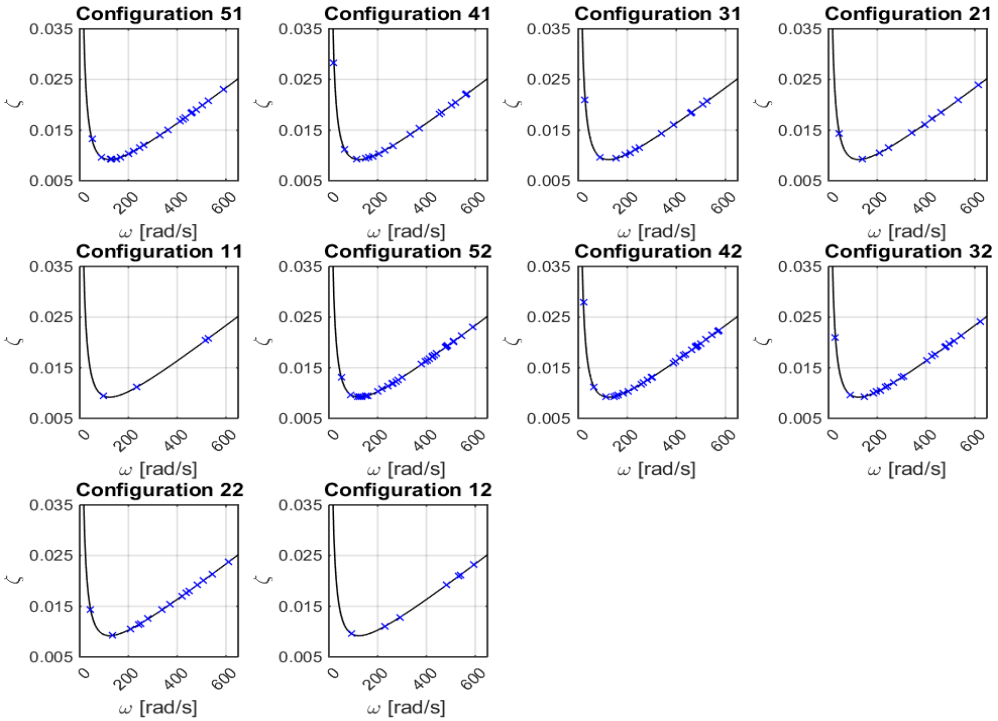

Figure 6.6: Damping ratio for all modes for 2D frames with section spans 6m modelled with 8-node plane stress elements (CPS8R) and material/global Rayleigh damping. First 10 of 19 frame configurations. Data plotted towards the Rayleigh curve.

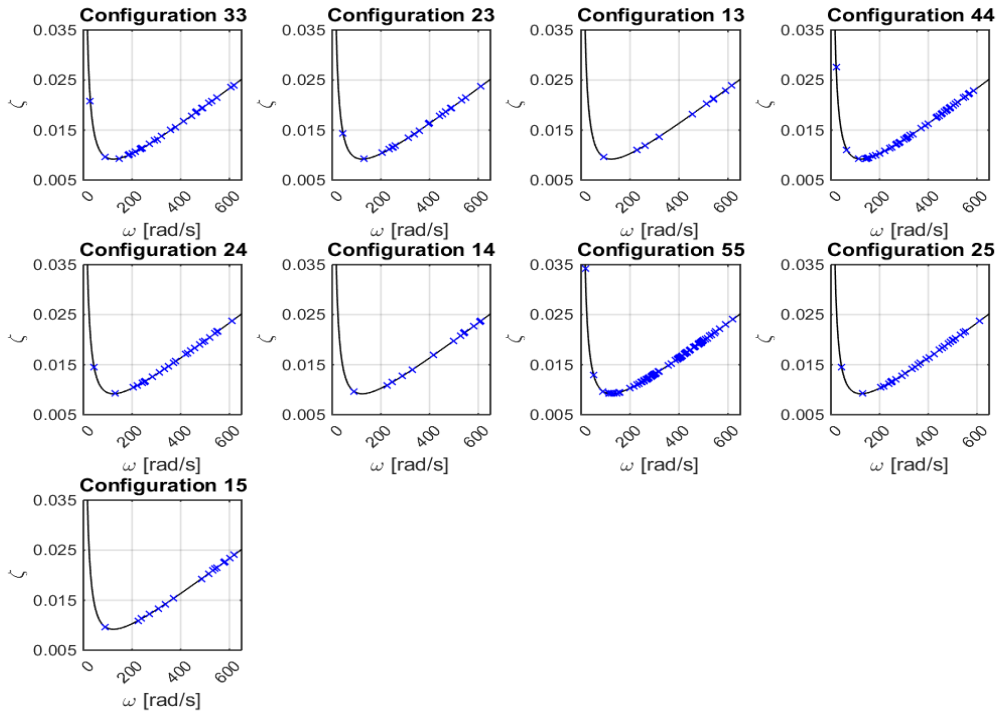

Figure 6.7: Damping ratio for all modes for 2D frames with section spans 6m modelled with 8-node plane stress elements (CPS8R) and material/global Rayleigh damping. Last 9 of 19 frame configurations. Data plotted towards the Rayleigh curve.

6.2.3 2D frames – Timoshenko beam elements (B21) – global vs material Rayleigh damping

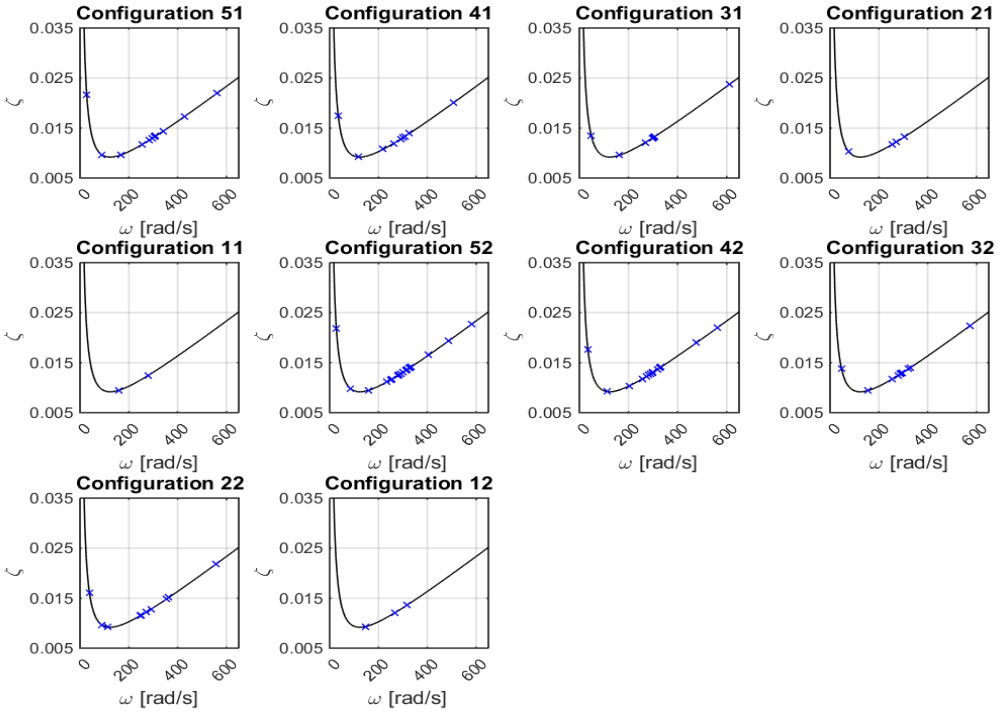

Figure 6.8: Damping ratio for all modes for 2D frames with section spans 6m modelled with Timoshenko B21- beam elements and global Rayleigh damping. First 10 of 19 frame configurations. Data plotted towards the Rayleigh curve.

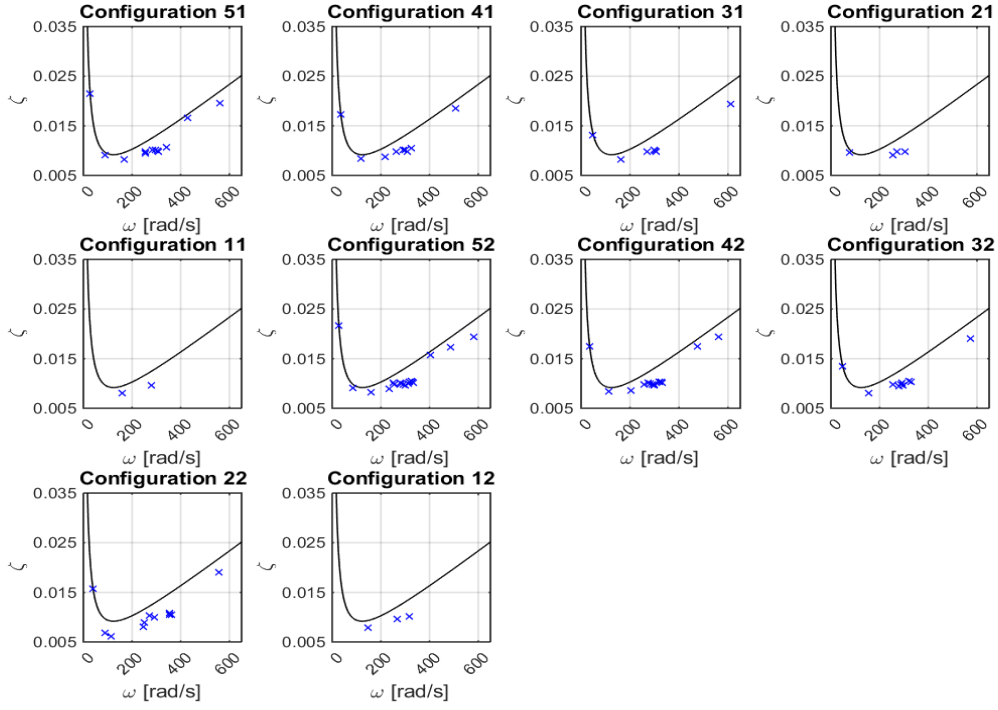

Figure 6.9: Damping ratio for all modes for 2D frames with section spans 6m modelled with Timoshenko B21- beam elements and material Rayleigh damping. First 10 of 19 frame configurations. Data plotted towards the Rayleigh curve.

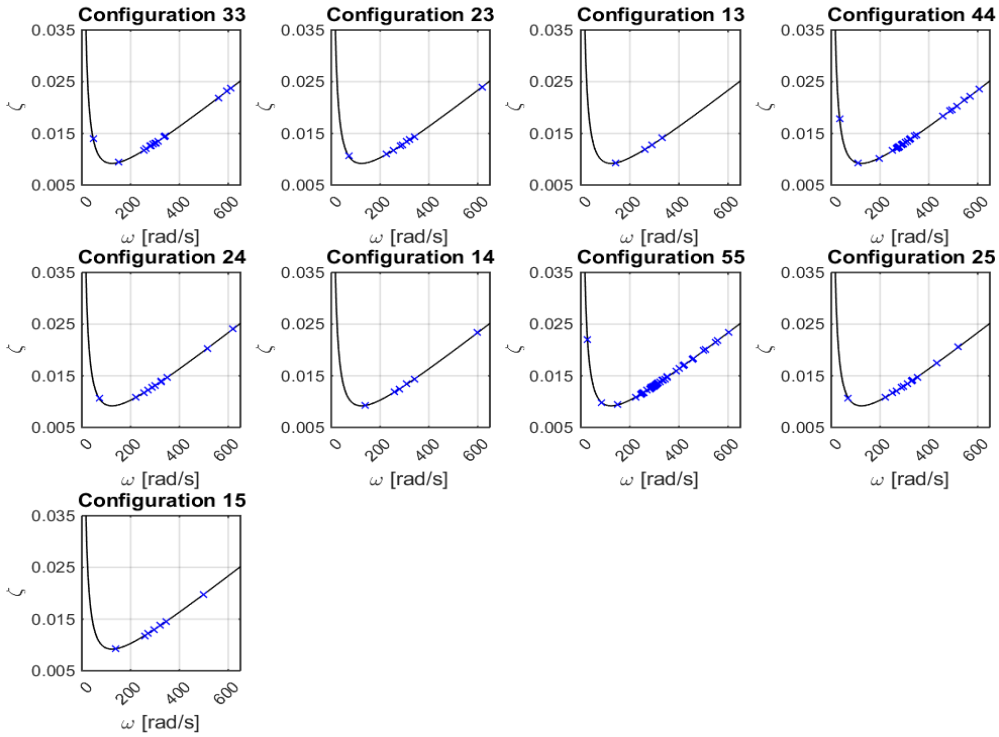

Figure 6.10:Damping ratio for all modes for 2D frames modelled with Timoshenko B21-beam elements and global Rayleigh damping. Last 9 of 19 frame configurations. Data plotted towards the Rayleigh curve.

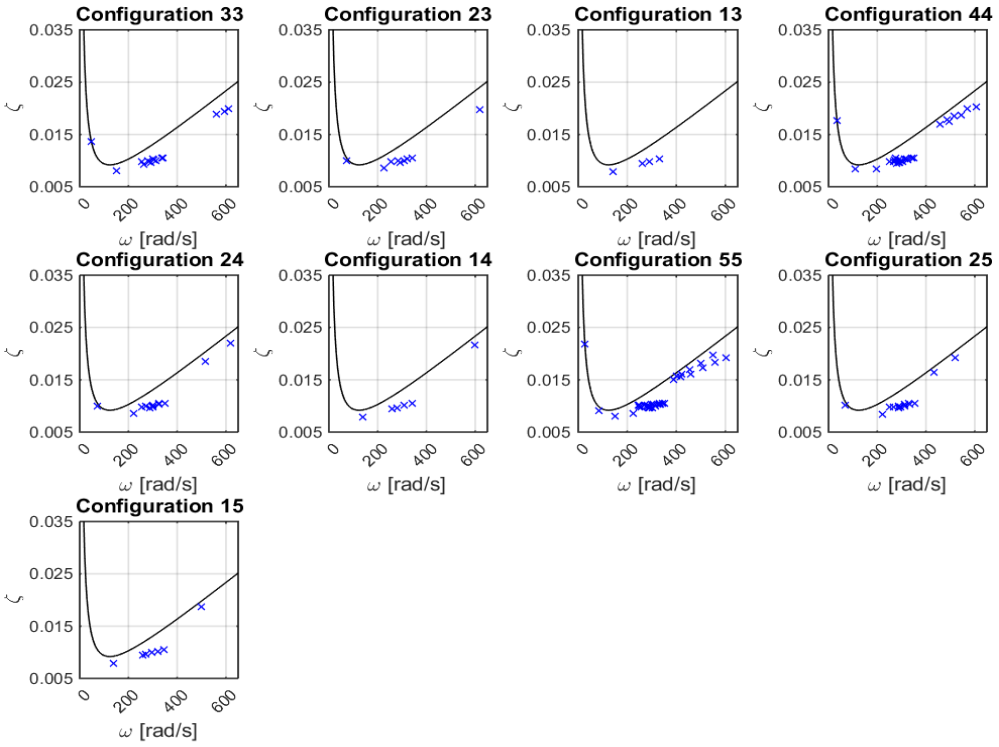

Figure 6.11: Damping ratio for all modes for 2D frames modelled with Timoshenko B21-beam elements and global Rayleigh damping. Last 9 of 19 frame configurations. Data plotted towards the Rayleigh curve.

Figure 6.8, Figure 6.9, Figure 6.10 and Figure 6.11 display the damping ratio, plotted towards the Rayleigh curve, for all calculated modes for all frame configurations with section spans of 6 meter, and modelled with Timoshenko 2D beam elements (B21). Every other figure display damping modelled as respectively global Rayleigh damping and material Rayleigh damping.

Figure 6.8 and Figure 6.10 show that global Rayleigh damping coincide with the mathematical Rayleigh curve, whereas figures Figure 6.9 and Figure 6.11 show that material Rayleigh damping deviates from the Rayleigh curve. These deviations are apparent for all frame configurations for all eigenfrequencies above 200 rad/s (~30 Hz) and are particularly large in the frequency ranges 300-350 rad/s (~50 Hz) and close to 600 rad/s (~100 Hz), approximately. The trend is larger deviations towards the Rayleigh curve for increasing frequencies. However, at frequencies 370-380 rad/s a "jump" in damping ratio appears, after which the calculated damping at first lay closer to the Rayleigh curve until the deviations again increase for increasing frequencies, but at a lower rate than before the damping ratio jump.

Although damping ratios are not the same for B21-element models with material damping and equivalent models with global damping, the calculated eigenfrequencies may at first sight seem to be identical by visual inspection of the plots. This is however disproved when investigating the data sets. Although differences are diminishingly small, the frequencies are in fact not numerically identical. The discrepancies between global Rayleigh damped and material Rayleigh damped eigenfrequencies is likely a result of the damping ratios being different. Since the complex eigenfrequencies consider damping and damping has an influence on damped eigenfrequencies.

In general, most eigenfrequencies are found for the highest frames. For the lowest frame configurations (less than three floors) few if any frequencies are found above 400 rad/s (65 Hz).

6.2.4 Summary of 2D frame results

The results with the negative damping ratios for the plane stress elements when global Rayleigh damping was added with SIM architecture was surprising. A negative damping ratio, along with positive real parts of the eigenvalues, indicate an unstable system. This is not good. An answer to why this happened for the plane stress model has not been found but could be investigated further.

Results comparing global and material Rayleigh damping for all frame models, both Timoshenko (B21) and plane stress (CPS8R) element models, with global damping, displayed damping ratios coinciding with the Rayleigh curve, as is expected from the mathematical Rayleigh damping formulation. This was also the case for the plane stress element models with material damping, whereas the Timoshenko element models with Rayleigh damping in general displayed deviations towards the Rayleigh curve. Thus,

there seem to be an analogy to the results for the beam models with the corresponding 3D-expansion of these elements, respectively B21 to B31 for 2D to 3D Timoshenko element and CPS8R to C3D20R for plane stress to volume element. Damping ratios for the Timoshenko B31-beam element models with material Rayleigh damping, showed discrepancies compared to the Rayleigh curve, while the results for the equivalent 3D solid element models coincided with the mathematical curve.

The fact that eigenfrequencies for the same B21-element frame configurations are not exactly numerically identical and depends on whether damping is implemented as material damping or global damping, indicates that the deviations for the Timoshenko element models with material damping lies in the calculations of the damping ratio. This is because the complex frequency step in Abaqus calculates the damped eigenfrequencies, which theoretically depend directly on the damping ratio for the mode. Although eigenfrequency discrepancies are small (and vibration modes are not visibly distinguishable by eye inspection), this coincide with theory, since frequency dependency on damping is weak for low levels of damping, which is the case here.

However, in mathematical terms, when only one material is used in the model the Rayleigh damping model should be identical between global damping specification and for specification of damping onto the material definition. The authors have not found obvious or certain causes for the material damping version to deviate. Followingly, neither have any causes been found for the observable “jump” in damping ratio for frequencies above approximately 375 rad/s.

One probably far-fetched hypothesis is that the discrepancies for material Rayleigh damping is related to the formulation of the Timoshenko elements in Abaqus. The damping ratios for higher frequencies with B21- and B31-elements are observed to either decrease, increase at slower rates or plateau at specific short-ranged intervals of damping levels. In general, with these elements it is seen from both the iSight simulations of the beams and from the 2D frames that the damping ratios are lower than they are expected to be when comparing towards the Rayleigh curve. Similar observations, that damping ratio is frequency independent or is decreasing for increasing eigenfrequency, are reported by Humar [5]. The far-fetched hypothesis is thus that Abaqus has implemented a behaviour that better matches experimental measurements. This “experimental behaviour” could be triggered when Rayleigh damping is implemented as material damping instead of global damping; one could assume that the implementation is meant to simulate the behaviour of a physical material in a somewhat better manner.

If this were the actual case, one could expect it to be the case for the shell elements and the 3D solid elements as well; however, it does not seem to be. From the iSight simulations these elements showed to decrease in both frequency and damping ratio for increased beam height, so that they stay true to the Rayleigh-curve. As a long shot one could think that these elements are computationally costly and therefore do not have the

more advanced “experimental behaviour” properties of the B21/B31-elements. What is known for the B21/B31-elements is that they are efficient elements for both slender and thick beams, which normally require either Euler-Bernoulli theory or Timoshenko theory. So, these elements are at least based on a very advanced formulation. Nevertheless, this hypothesis is a long shot and seems unlikely.

6.3 Floor element

A floor element has been tested and analysed in Abaqus. A brief description of this floor element is presented here:

- Spans 9,0x2,4 m
- An enclosed frame of glulam beams makes up the outer edge, cross section is 140x405 mm (longitudinal beams are 9 000 mm long and transversal are 2 120 mm)
- Three longitudinal glulam beams with cross section 66x405mm evenly distributed inside the frame
- Top and bottom made from LVL plates glued to the beams, which respectively are 45 mm and 63 mm thick

6.3.1 Modelling the floor element

The floor element was modelled with shell elements for the top and bottom plate, and 3D solid elements for the beams. The different components in the floor element was made as parts and assembled as instances. These parts were made according to the description in section 4.2. Deviations from section 4.2 are made to have correct elastic features. The materials used are described below in Table 6.1. KertoQ is for the plates, GL28c for the smaller beams inside the outer frame, and GL30c for the bigger beams that make up the frame.

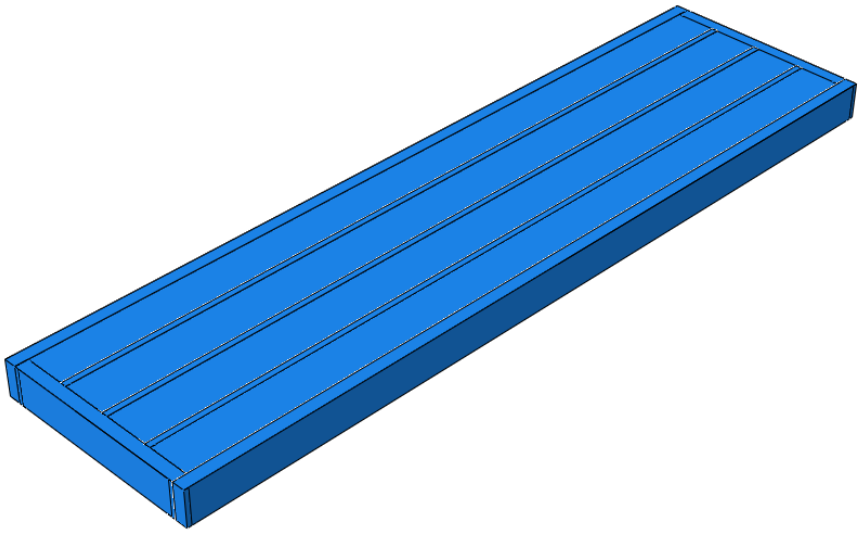

Figure 6.12: A floor element was modelled in Abaqus. Here are the different instances positioned to make up the floor element.

The instances were positioned to make up the floor element and tied together by adding constraints of the type "tie". An illustration is presented in Figure 6.12. For the tying of the surfaces it is important that the slave surface has a higher density of elements than the master surface. Tests revealed discrepancies in mode shapes if the master/slave surfaces were chosen incorrectly. If done incorrectly Abaqus will output a warning message.

In the physical floor element, the beams are screwed together by densely spaced screws and the plates are glued to the beams. The glue is assumed stronger than the material to be glued and the screws are so densely spaces that in the model all the connecting surfaces are assumed tied/fixed together.

Table 6.1: Material definitions for the floor element. Features referring to Abaqus material module. [21] [22] [23]

Material	GL28c	GL30c	KertoQ
E1 [N/mm^2]	12 500	13 000	10 590
E2 [N/mm^2]	300	300	2 967
E3 [N/mm^2]	300	300	244.6
Nu12 [ ]	0.02	0.02	0.02
Nu13 [ ]	0.02	0.02	0.02
Nu23 [ ]	0.3	0.3	0.68
G12 [N/mm^2]	650	650	500
G13 [N/mm^2]	650	650	147.5
G23 [N/mm^2]	65	65	48.9
Density [kg/m^3]	420	430	510

6.3.2 Analysis of modes and damping

Different elements were tested for the floor element. One model was made with only linearly interpolated elements (in Abaqus referred to as C3D8R for the 3D solid elements and S4R for the shell elements). This model yielded 95 modes below 100 Hz. From these, there were 18 with an effective damping ratio in the order 10^{-2}. The rest of the modes had effective damping ratios in the order 10^{-3}. These modes were visibly separable from those with higher damping: they were local buckling modes for the slender, internal beams, i.e. the whole floor element did not vibrate in a global sense. An example of this can be seen in Figure 6.13. For the modes with effective damping ratios in the order 10^{-2} there were some degree of buckling of the outer larger beams.

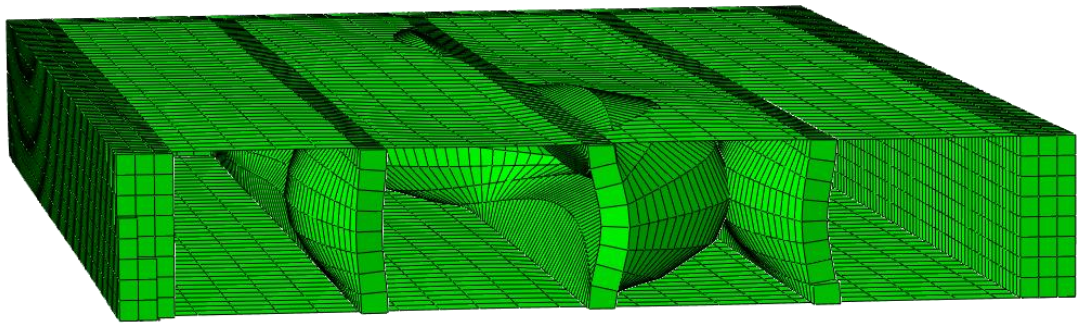

Figure 6.13: A cross sectional view through the middle of the floor element. Showing how the slender beams in the centre buckle locally.

Comparing the reported damping for the different modes, some match the Rayleigh-curve (the same curve as discussed in the previous chapters) to varying degree and some do not match at all. The modes with effective damping ratio in the order 10^{-3} are highly mismatched with the Rayleigh-curve. All modes up to about 50 Hz match quite well, and those above 50 Hz with effective damping ratio in the order 10^{-2} are all below the Rayleigh curve, but match to some degree.

For a second model, all the elements where quadratically interpolated (the 3D solid element was C3D20R and the shell element was S8R). This model displayed more reasonable results. Below 100 Hz, 12 modes where found and none of these where local buckling modes, meaning that all the modes displaced the whole floor element to some degree. All the modes match with the Rayleigh curve to some degree. The best matches are found for modes below 80 Hz.

When comparing the frequency and shape of the modes between the model with linear elements and the model with quadratic elements there are little similarities to be found. A few modes are similar but are found at different frequencies, and there are types of modes that only the quadratic model report. These are horizontal modes, higher order of vertical modes and full warping/twisting of the floor element.

A third and last model was made with linear shell elements (S4R) and quadratic 3D solid elements (C3D20R). This model had a lot of modes similar to the modes of the quadratic

model, but at different frequencies. E.g. the first vertical beam-like bending mode was at 13 Hz for the third model and at 30 Hz for the second model. The number of modes and the magnitude of the effective damping ratios was also like the quadratic model.

The great number of local modes in the first model was assumed to be a problem with hourglassing, which can appear when applying reduced integration and linear elements [24]. Further, when the whole model was "upgraded" to quadratic elements, this had different issues. The quadratic version of the shell element is based on Mindlin-Reissner plate theory, which are best suited for thick shell applications [7]. A rough and common limit between thick and thin plates is a length to thickness ratio (L/t) of 10. The bottom plate of the floor element is thickest and is 63 mm thick. When considering the shortest span length, the bottom plate has a ratio of about 38 ($\frac{2400}{63} \approx 38$), which is almost four times thinner than a plate in between thick and thin. Following this, the quadratic shell element (S8R) is not suited and will have issues with shear locking, thus, the model will appear stiffer than it should [24].

After changing the plates of the model to linear shell elements (S4R elements), which are best suited for thin plates, the lowest eigenfrequencies became lower than the model with all quadratic elements. A specific example is the first vertical mode that came down from 30 Hz to 13 Hz, which shows it was likely a problem with shear locking for the thick shell elements.

6.4 3D frame with floor element

The most complex structure modelled in this work was a 3D frame with a floor consisting of the floor element discussed in the previous section. This model was just briefly analysed. It was modelled as a test to see how the modelling techniques worked in 3D and to see if structural damping could be added to parts of the structure were the connection between different parts of the structure are present.

The model is displayed in Figure 6.14. Here one can see small parts on the ends of the outer beams in the floor element and parts in columns were the floor element is connected. These parts are made by partitioning in Abaqus so that these parts can be represented by a different material than the rest of the model. The model was tested by having a material with Rayleigh damping in the whole model except from the regions where the floor element is connected to the columns. These regions have a material that have all the same properties and additionally some structural material damping.

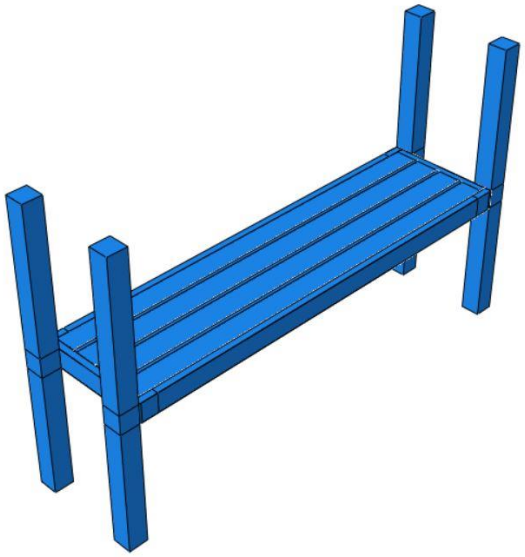

Figure 6.14: 3D frame model with floor element.

Results from this analysis showed that when the structural damping was added to these specific parts the damping was increased. This shows that the structural damping and the Rayleigh damping can be combined to make a more adequate model. The results were not investigated any further than this.

7 Summary

7.1 Conclusion

The iSight-simulations with material damping showed expected results for all the Euler-Bernoulli beams, expected in the sense that the eigenfrequencies matched the analytical expression from Euler-Bernoulli beam theory and the damping ratios matched the Rayleigh curve. As a result, one can predict what levels of damping that will be present in a model with the EB beam elements. This was not the case for the three elements that include shear deformations: the Timoshenko beam elements (B31), the thick shell elements (S8R) and the 3D solid elements (C3D20R). For these elements, damping ratios were found to decrease for beam slenderness values below approximately 6, even though they were expected to increase.

All the discrepancies seemed to appear for beam thicknesses for which the shear stiffness presumably will influence the results. Tests with varying shear stiffness showed that the rolling shear stiffness had no influence. The two longitudinal stiffnesses had an influence independent of each other, corresponding to the direction of the vibration mode. However, the influence was minor, and the shear stiffness does not fully explain the discrepancies.

The eigenvalue problem was solved analytically with Timoshenko elements. When increasing the number of elements, the eigenfrequencies converged towards the Abaqus solutions with Timoshenko B31-elements. This indicates that eigenfrequency calculations with Timoshenko elements in Abaqus comply with theory. Since Timoshenko theory should be accurate for shear-flexible beams, this further strengthens the suspicion that the two other shear-flexible elements – the shell and 3D solid – results in underestimated eigenfrequencies for the thickest beams.

Mode shapes were investigated by the modal assurance criterion (MAC). For beam elements and shell elements with shell thickness as beam height, MAC did not reveal any such effect, although correlation was found in cases where this was expected. For 3D solid element models and shell element models with shell thickness as beam width, MAC did find mismatch in mode correlation in the expected cases, that possibly could be caused by different levels of shear deformations. However, due to weaknesses related to the MAC formulation, as well as non-consistent results between element types, MAC was considered a method to be more indicative than conclusive. Presumably, MAC works best as a supplement to other methods.

Since Rayleigh damping is proportional to the mass and stiffness matrices, these were investigated. Compared to the analytically established matrices, the mass matrix was compliant, but the stiffness matrix was not. The B31-Timoshenko element in Abaqus is formulated to be a general-purpose element, thus presumably having a quite complex

formulation. This could be one reason to why the analytical matrix does not match the matrix outputted from Abaqus.

For both simply supported beams in 2D and 3D and 2D frames, the shear flexible beam elements displayed discrepancies in damping ratios towards the Rayleigh curve. When SIM architecture was applied the global Rayleigh damping was not reliable, but for traditional architecture it matched perfectly with the Rayleigh curve. Nevertheless, the material Rayleigh damping were poorer when using the traditional architecture. With plane stress elements the results from material Rayleigh damping were perfectly matched.

The more complex 3D frame model with a floor element, showed that a combination of material Rayleigh damping and structural damping works fine. In this way, one could without difficulties also define different levels of Rayleigh damping and structural damping in different parts of the structure. A remark should be made that when combining different levels of damping at the material level, the distribution of the different materials has an influence on the outputted damping ratio.

7.2 Further work

Following are the suggestions for further work:

- Investigate why material Rayleigh damping with SIM architecture lead to poor results for some elements.
- Measure the material damping from experimental tests.
- Measure the damping from typical joints and connections.
- Investigate damping through energy parameters.

If one could discover and resolve the issue with SIM architecture and material Rayleigh damping, one could have more options for damping. A comprehensive damping model that combine Rayleigh damping at the material and global level with structural damping at the material level could be applied.

There is not enough data on how much material damping different timber materials contribute with in a model. If one had data for the most used components with different material configurations, one could adjust the input in the numerical model and possibly describe damping more accurate. The same is also the case for typical joints and connections for timber structures.

Influence on damping from shear deformations could be investigated by studying energy parameters available in other dynamic procedures, since the complex frequency step does not include such.

8 References

[1] DynaTTB consortium, «DynaTTB,» 2019. [Internett]. Available: https://www.dynattb.com/. [Funnet 6 Juni 2020].

[2] N. Labonnote, «Damping in Timber Structures,» Norwegian University of Science and Technology, Trondheim, 2012.

[3] C. W. Edited by: De Silva, i *Vibration and Shock handbook*, Boca Raton, Florida, Taylor and Francis group, 2005, pp. 3: 32-35; 9: 27-35; 19:.

[4] A. K. Chopra, Dynamics of Structures Theory and Applications to Earthquake Engineering, Upper Saddle River: Prentice Hall, 2012.

[5] J. L. Humar, Dynamics of structures, London: Taylor & Francis Group, 2012.

[6] S. S. Rao, Vibration of Continuous Systems, Hoboken: John Wiley & Sons, 2007.

[7] Dassault Systèmes SIMULIA corp., «Abaqus Theory Guide,» *Abaqus 6.14 Documentation,* nr. Abaqus 6.14, 2014.

[8] Dassault systèmes SIMULIA Corp., «Abaqus Analysis User's Guide,» *Abaqus 6.14 Documentation,* nr. Abaqus 6.14 Documentation, 2014.

[9] N. E. A. Rønnquist, *TKT4201 - Lecture 9: Damping of structures,* Trondheim: NTNU, 2019.

[10] D. J. Ewins, Modal testing: theory, practice and application, Baldock: Research studies press LTD., 2000.

[11] T. M. (Siemens), «Siemens simcenter,» 29 aug 2019. [Internett]. Available: https://community.sw.siemens.com/s/article/modal-assurance-criterion-mac. [Funnet 01 Mars 2020].

[12] A. H. Reigstad og J. L. Sandnes, «A preliminary study on modelling of material damping for timber structures,» NTNU, Trondheim, 2019.

[13] C. W. de Silva, Vibration Fundamentals and Practice, Boca Raton: CRC Press, 2000.

[14] Standard Norge, «Structural timber - Strength classes,» Standard Norge, 2016.

[15] F. Mirianon, F. Stefania og T. Toratti, « A method to model wood by using ABAQUS finite element software,» VTT Technical Research Centre of Finland, Espoo, 2008.

[16] K. A. Malo, *Summary: Timber Structure II - TKT4212,* Trondheim: NTNU, 2019.

[17] K. Bell, Matrisestatikk Statiske beregninger av rammekonstruksjoner, Fagbokforlaget Vigmostad & Bjørke AS, 2011.

[18] R. D. Cook, D. S. Malkus og M. E. Plesha, Concepts and applications of finite element analysis, Madison: John Wiley & Sons, 1989.

[19] K. Bell, Konstruksjonsmekanikk del II Fasthetslære, Bergen: Fagbokforlaget Vigmostad & Bjørke AS, 2015.

[20] MIT, «Damping,» 21 September 2016. [Internett]. Available: https://abaqus-docs.mit.edu/2017/English/SIMACAEKEYRefMap/simakey-r-damping.htm. [Funnet 30 April 2020].

[21] Tlustochowicz og Gabriela, «Stabilising System for Multi-Storey Beam and Post Timber Buildings,» Luleå Univeristy og Technology, Luleå, 2011.

[22] Standard Norge, «Trekonstruksjoner Limtre og limtlaminert heltre Krav,» Standard Norge, 2013.

[23] VTT Expert Services LTD, «VTT Certificate NO. 184/03,» VTT Expert Services LTD, 2004.

[24] K. Bell, An engineering approach to finite element analysis of linear structural mechanics problems, Bergen: Fagbokforlaget Vigmostad & Bjørke AS, 2013.

[25] W. Zhang, M. Vinayagamoorth and L. Duan, "Dynamic Analysis," 24 January 2014. [Online]. Available: https://www.routledgehandbooks.com/doi/10.1201/b15663-4. [Accessed 17 December 2014].

[26] C. Brown, B. Hartz og C.-T. Yeh, «Damping sources in wood structures,» Jouran of Sound and Vibration, 1971.

[27] R. J. Allemang, «The Modal Assurance Criterion - Twenty Years of Use and Abuse,» *Sound and vibration,* August 2003.

Appendix

Appendix A: Example MATLAB script for calculation of analytical system matrices and solving the EVP

```
% EXAMPLE ON ESTABLISHING THE SYSTEM STIFFNESS MATRICES BY THE DIRECT
% STIFFNESS METHOD,
% ON SOLVING THE EIGENVALUE PROBLEM,
% AND ON CALCULATING RAYLEIGH COEFFICIENTS FOR INPUT DAMPING RATIOS.
%
% System stiffness matrices are established by the direct stiffness method
% for a 2-element-system with either EB or Timoshenko elements. EVP is
% solved and modal transformation of system matrices are performed with
% undamped eigenvectors.
%
% Modelled by either:
% Euler-Bernoulli beam elements (2 elements; in Abaqus CAE: B23), or
% Timoshenko beam elements (2 elements; in Abaqus CAE: B21).
% Follow instructions below to choose element type, i.e. to include or
% exclude shear deformations.
clc, clear all
format bank %use to write out, i.e. avoid exponential format

%Beam specifications:
%density, rho = 420kg/m^3; length=choose;  width = 0,1m; height=choose

% INPUT PARAMETERS:
rho = 420; % [kg/m^3]
L = 3; %NB! length of element is beam length/2
w = 0.1; h = 0.2;
E = 11e+9; %E-modulus, E = 11000 N/mm^2 = 11GN/m^2
G = 690e+6; %shear stiffness (longitudinal/ out of plane), (from EN338)
%G = 3928.6e+6; %shear stiff (longit) isotropic model [N/mm^2=MN/m^2]


% CALCULATED PARAMETERS
I = (w*h^3)/12; %2nd moment of inertia
A = w*h; %cross-sectional area
kappa = 6/5; %shear coefficient related to cross-seciton geometry
As = A/kappa; %shear area
m = rho*w*h; %mass per beam length [kg/m]
a_s = (12*E*I)/(G*As*L^2); %shear factor/ coefficient

% % Element stiffnes matrix (NO SHEAR DEFORMATIONS INCLUDED)
% % "%" out if shear included:
% ki = E*I/L^3*[12 -12 6*L 6*L;
%               -12 12 -6*L -6*L;
%               6*L -6*L 4*L^2 2*L^2;
%               6*L -6*L 2*L^2 4*L^2];

% Element stiffness matrix (INCLUDING SHEAR DEFORMATIONS):
% "%" out if shear not included in Abaqus analysis
ki = E*I/((1+a_s)*L^3)*[12 -12 6*L 6*L;
             -12 12 -6*L -6*L;
             6*L -6*L (4+a_s)*L^2 (2-a_s)*L^2;
             6*L -6*L (2-a_s)*L^2 (4+a_s)*L^2];
```

```
% Element mass matrix
mi = m*L/420*[156 54 22*L -13*L;
              54 156 13*L -22*L;
              22*L 13*L 4*L^2 -3*L^2;
              -13*L -22*L -3*L^2 4*L^2];

% Connectivity matrices for 2-element beam:
a1 = [0 0 0 0;
      0 1 0 0;
      1 0 0 0;
      0 0 1 0];
a2 = [0 1 0 0;
      0 0 0 0;
      0 0 1 0;
      0 0 0 1];

% SYSTEM stiffness- and mass matrces:
K = a1'*ki*a1 + a2'*ki*a2
M = a1'*mi*a1 + a2'*mi*a2
% % check element stiffness- and mass matrices element 1
% K = a1'*ki*a1
% M = a1'*mi*a1
% % check element stiffness- and mass matrices element 2
% K = a2'*ki*a2
% M = a2'*mi*a2

%Solving eigenvalue problem for the system (NB! shear included or not?):
[eigVec, eigVal] = eig(K,M);
wn2 = diag(eigVal);
PHI = eigVec; %mode shape matrix (vibration modes)

%Natural frequencies:
wn = sqrt(wn2); %Natural (circular) frequencies of the system [rad/s]
fn = wn/(2*pi) %Natural frequencies of the system [Hz]

% modal transformation of system:
%orthogonalizing/ generalizing the system (decoupling the system) into the
%modal stiffness- and modal mass matrices to get modal (system) damping
%matrix expressed with alpha and beta --> solve for modal damping ratios:

KS = PHI'*K*PHI; %Modal stiffnes matrix
MS = PHI'*M*PHI; % Modal mass matrix
% See under for modal damping matrix

%choosing modes of vibration, n and m, to solve for modal damping
%ratios zeta_i:
n = 1; m = 2; %check/ specify before running.
%CHOSING MODAL DAMPING RATIOS FOR THE TWO FIRST VIBRATION MODES, MODE 1
%AND MODE 2: zeta_1 (n=1) and zeta_2 (m = 2):
% (Magnitude of damping ratios of same order as SW-E-5-1 &
% SW-E-5-2 beam configurations in Labonnote's Paper I, Table 4
% (SW-E-5-1 = 0.0095, SW-E-5-2 = 0.0149)):
zeta_1 = 0.01;
zeta_2 = 0.015;

% CALCULATING RAYLEIGH DAMPING COEFFICIENTS BASED ON CHOSEN MODAL DAMPING
% RATIOS FOR FIRST AND SECOND MODE
R_coeffs = 2*inv([1/wn(n) wn(n); 1/wn(m) wn(m)])*[zeta_1; zeta_2];
alpha = R_coeffs(1);
beta = R_coeffs(2);
```

Appendix B: Analytical calculation of eigenfrequencies and comparison to Abaqus-calculated eigenfrequencies for the shear-flexible B31-element.

```
% READING IN AND PROCESSING DATA FROM EXCELL SPREADSHEET FROM ISIGHT
% SIMULATION:
% Note: inpdata must be spreadsheet from Isight simulation;
% Copy directly the .txt-file storing the results from Isight into an excel
% spreadsheet file and save as either "excel 97-2003 regneark" (.xls)
% or "excel regneark" (.xlsx).
clc, clear all, close all

%% Reading excel spreadsheet, by xlsread-function, into a variable:

% "xlsread" returns the numeric data in input array,
% i.e. this option inly reads in numeric data from the spreadsheet, so
% excludes the first row containing the variable names for each column,
% and excludes the two first columns containing strings

inpdata = xlsread('B31vf_elsz50.xlsx');
%Fill in name Abaqus model without underscores, "_":
Name = 'B31 plot omega analytEB vs EVP 5xT vs AbaqusB31';
%NB: must adjust: parameter "lengths", T or EB stiffness mat K,
%and compatibility matrices corresponding to #elements included, and
%legend for EVP with corresponding element and #elements.
%--------------------------------------------------------------------------

%% Extracting wanted data from main variable
% Note that Isight write value 1000 as "1 000" with a space, so it is
% regarded NaN in matlab.
[inputRows, inputCols] = size(inpdata);
heightsAllConfigs = inpdata(:,3);
heightsAllConfigs(20:20:end) = 1000; %fix NaN values
h = heightsAllConfigs(1:20);
beamConfigs = [1*ones(20,1); 2*ones(20,1); ...
     3*ones(20,1); 4*ones(20,1); 5*ones(20,1)];

data = inpdata(:,5:2:end);
[drows, dcols] = size(data);

if dcols < 2
    disp('Not enough input data')
elseif dcols == 2
    disp('Only 1st natural frequency found for all configs')
    firstEDRs = data(:,1);
    firstFreqs_f = data(:,2);
elseif dcols == 4
    disp('1st and 2nd natural frequencies found for some congfigs')
    firstEDRs = data(:,1);
    secondEDRs = data(:,2);
    firstFreqs_f = data(:,3);
    secondFreqs_f = data(:,4);
elseif dcols == 6
    disp('Three first natural frequencies found for some configs')
    firstEDRs = data(:,1);
    secondEDRs = data(:,2);
    firstFreqs_f = data(:,4);
    secondFreqs_f = data(:,5);
```

```
elseif dcols == 8
    disp('Four first natural frequencies found for some configs')
    firstEDRs = data(:,1);
    secondEDRs = data(:,2);
    firstFreqs_f = data(:,5);
    secondFreqs_f = data(:,6);

elseif dcols == 10
    disp('Five first natural frequencies found for some configs')
    firstEDRs = data(:,1);
    firstFreqs_f = data(:,6);
    secondEDRs = data(:,2);
    secondFreqs_f = data(:,7);
elseif dcols == 12
    disp('Check excel file if actually 6 modes... Remove excess data')
end %outer if
%-------------------------------------------------------------------------
%% f --> omega, for plotting, and separating data for beam configs:
if dcols == 2
DRs_1 = firstEDRs/2; %damp rat = EDR/2, approximately
DR_1_1m = DRs_1(1:20);
DR_1_2m = DRs_1(21:40);
DR_1_3m = DRs_1(41:60);
DR_1_4m = DRs_1(61:80);
DR_1_5m = DRs_1(81:100);

first_w = firstFreqs_f*(2*pi);
omega_1_1m = first_w(1:20);
omega_1_2m = first_w(21:40);
omega_1_3m = first_w(41:60);
omega_1_4m = first_w(61:80);
omega_1_5m = first_w(81:100);

elseif dcols >= 4
DRs_1 = firstEDRs/2; %damp rat = EDR/2, approximately
DR_1_1m = DRs_1(1:20);
DR_1_2m = DRs_1(21:40);
DR_1_3m = DRs_1(41:60);
DR_1_4m = DRs_1(61:80);
DR_1_5m = DRs_1(81:100);
DRs_2 = secondEDRs/2; %damp rat = EDR/2, approximately
DR_2_1m = DRs_2(1:20);
DR_2_2m = DRs_2(21:40);
DR_2_3m = DRs_2(41:60);
DR_2_4m = DRs_2(61:80);
DR_2_5m = DRs_2(81:100);

first_w = firstFreqs_f*(2*pi); %f [Hz] --> circular frequency [rad/s]
omega_1_1m = first_w(1:20);
omega_1_2m = first_w(21:40);
omega_1_3m = first_w(41:60);
omega_1_4m = first_w(61:80);
omega_1_5m = first_w(81:100);
sec_w = secondFreqs_f*(2*pi); %f [Hz] --> circular frequency [rad/s]
omega_2_1m = sec_w(1:20);
omega_2_2m = sec_w(21:40);
omega_2_3m = sec_w(41:60);
omega_2_4m = sec_w(61:80);
omega_2_5m = sec_w(81:100);
end %if
%-------------------------------------------------------------------------
```

```
%% Plotting data
%%
max_freq_Hz = 100;
max_circular_freq = max_freq_Hz*2*pi; %Max --> 628Hz
%Rayleigh-curve parameters:
w = [5:5:700];
alpha = 1.13; beta = 7.47e-5;
z = alpha./(2*w) + (beta.*w)/2;

%% Analytical natural undamped frequencies and damped frequencies calc.
% NB! Use correct 2nd moment of area for respective modes, change below
%beam parameters:
W = 0.1; %beam width [m]
L = [1 2 3 4 5]; %[m] beam lengths
heights = [.050 .100 .150 .200 .250 .300 .350 .400 .450 .500 .550 ...
    .600 .650 .700 .750 .800 .850 .900 .950 1.0]; %[m] beam heights
rho = 420; %[kg/m^3], density
E = 11e+9; % [N/m^2]
% Second moment of area about strong axis --> vertical modes:
I = (1/12)*W*(heights.^3); %[m^4], second moment of area, vertical modes
% Second moment of area about weak axis --> horizontal modes:
% I = (1/12)*heights*(W^3); %[m^4], second moment of area,horizontal modes
m = rho*W*heights; %[kg/m], distributed beam mass, mass per length
%CHOOSE VIBRATION MODE NUMBER BY SETTING VALUE OF n:
n = 1; %n is the vibration mode number, i.e. n=1 give first natural freq.

%ANALYTICAL UNDAMPED NATURAL FREQUENCIES ALL BEAM CONFIGURATIONS:
omega_n1 = zeros(length(heights), length(L));
for i = 1:1:length(L)
    omega_n1(:,i) = (((n*pi)/L(i))^2).*sqrt((E*I)./m);
end %for
omega_n1; %natural circular frequencies [rad/s] [20x5-matrix]
% w_n1 = [omega_n1(:,1); omega_n1(:,2); omega_n1(:,3); ...
%     omega_n1(:,4); omega_n1(:,5)]; % nat.circ.freq.s [100x1-matrix]
% f_n1 = omega_n1/(2*pi); %natural frequencies [Hz] [20x5-matrix]

% ALT. 1: DAMPED FREQUENCIES CALCULATION, ALL FREQUENCIES EQUALLY DAMPED:
zeta = .02; % damping level of 2%.
omega_d1 = omega_n1.*sqrt(1-zeta^2); %circular frequency [rad/s]
% f_d1 = f_n1*sqrt(1-zeta^2); %frequency [Hz]

% % ALT 2: DAMPED FREQUENCIES CALCULATION, FREQUENCIES DAMPED WITH
% % CORRESPONDING OUTPUT DAMPING RATIO FROM ABAQUS ANALYSIS:
% DR1 = [DR_1_1m DR_1_2m DR_1_3m DR_1_5m DR_1_5m];
% OMEGA_D1 = omega_n1.*sqrt(1-DR1.^2); %circular frequency
%-------------------------------------------------------------------------
%% Eigenfrequencies from EVP based on Timoshenko-element model 2 elements:
%Beam specification:
% density, rho = 420kg/m^3; length = 6m;  width = 0,1m; height = 0,2m
% INPUT PARAMETERS:
% rho = 420; %defined above
% L = 2.5; %NB! length of element is beam length/2
% w = 0.1; %defined above
% h = 0.2;
% E = 11e+9; %E-modulus, E = 11000 N/mm^2 = 11GN/m^2 %defined above
G = 690e+6; %shear stiffness (longitudinal/ out of plane) (EN338)

%-----> NB! MUST CHANGE #ELMNTS HERE ACC TO WANTED:
lengths = (1:1:5)/5; %change /j acc to number of elmnts: 2 or 5.
```

```
% heights = 0.05:0.05:1; %defined above

wn = zeros(length(lengths),length(heights));
for i = 1:length(lengths)
    L = lengths(i); %NB! L is now redefined compared to above...!
    for j = 1:length(heights)
% CALCULATED PARAMETERS
H = heights(j);
I = (W*H^3)/12; %2. mom. of inertia
A = W*H; %shear area
kappa = 6/5; %shear factor, kappa=6/5 for rectangular cross sections
As=A/kappa;
m = rho*W*H; %mass per beam length [kg/m] %NB: m is here redefined
a_s = (12*E*I)/(G*As*L^2); %shear factor/ coefficient

% % Element stiffnes matrix EB-elmnt (NO SHEAR DEFORMATIONS INCLUDED)
% % "%" out if shear included:
% ki = E*I/L^3*[12 -12 6*L 6*L;
%               -12 12 -6*L -6*L;
%               6*L -6*L 4*L^2 2*L^2;
%               6*L -6*L 2*L^2 4*L^2];

% Element stiffness matrix T-elmnt(INCLUDING SHEAR DEFORMATIONS):
% "%" out if shear not included in Abaqus analysis
ki = E*I/((1+a_s)*L^3)*[12 -12 6*L 6*L;
              -12 12 -6*L -6*L;
              6*L -6*L (4+a_s)*L^2 (2-a_s)*L^2;
              6*L -6*L (2-a_s)*L^2 (4+a_s)*L^2];

% Element mass matrix
mi = m*L/420*[156 54 22*L -13*L;
              54 156 13*L -22*L;
              22*L 13*L 4*L^2 -3*L^2;
              -13*L -22*L -3*L^2 4*L^2];

% % EVP SOLVED WITH 2 ELEMENTS:
% % Connectivity matrices:
% a1 = [0 0 0 0;
%       0 1 0 0;
%       1 0 0 0;
%       0 0 1 0];
% a2 = [0 1 0 0;
%       0 0 0 0;
%       0 0 1 0;
%       0 0 0 1];
%
% % SYSTEM stiffness- and mass matrces, 2-element system:
% K = a1'*ki*a1 + a2'*ki*a2;
% M = a1'*mi*a1 + a2'*mi*a2;

% EVP SOLVED WITH 5 ELEMENTS
% Connectivity matrices (elementdofs=[vz1 vz2 vtheta1 vtheta2]):
a1 = [0 0 0 0 0 0 0 0 0 0;
      0 1 0 0 0 0 0 0 0 0;
      1 0 0 0 0 0 0 0 0 0;
      0 0 1 0 0 0 0 0 0 0];
a2 = [0 1 0 0 0 0 0 0 0 0;
      0 0 0 1 0 0 0 0 0 0;
      0 0 1 0 0 0 0 0 0 0;
      0 0 0 0 1 0 0 0 0 0];
```

```
a3 = [0 0 0 1 0 0 0 0 0 0;
      0 0 0 0 0 1 0 0 0 0;
      0 0 0 0 1 0 0 0 0 0;
      0 0 0 0 0 0 1 0 0 0];
a4 = [0 0 0 0 0 1 0 0 0 0;
      0 0 0 0 0 0 0 1 0 0;
      0 0 0 0 0 0 1 0 0 0;
      0 0 0 0 0 0 0 0 1 0];
a5 = [0 0 0 0 0 0 0 1 0 0;
      0 0 0 0 0 0 0 0 0 0;
      0 0 0 0 0 0 0 0 1 0;
      0 0 0 0 0 0 0 0 0 1];

% SYSTEM stiffness- and mass matrces, 5-element system:
K = a1'*ki*a1 + a2'*ki*a2 + a3'*ki*a3 + a4'*ki*a4 + a5'*ki*a5;
M = a1'*mi*a1 + a2'*mi*a2 + a3'*mi*a3 + a4'*mi*a4 + a5'*mi*a5;

%Solving eigenvalue problem for the system (NB! shear included or not?):
[eigVec, eigVal] = eig(K,M);
wn2 = diag(eigVal);
%Natural frequencies from EVP:
wn(i,j) = sqrt(wn2(1)); %Natural (circl) frequencies of the system [rad/s]
    end
end
wn;
%--------------------------------------------------------------------------

%% Plot of frequency as function of beam height
% frequency as function of beam height compared to
% analytical frequencies.

if dcols == 2
% should not be the case
elseif dcols >= 4
% FIGURE: Frequency 1st mode for respective beam heights
figure(1)
% Euler-Bernoulli based formula -  analytical undamped frequencies:
plot(h, omega_n1(:,1), 'co', h, omega_n1(:,2), 'go', ...
    h, omega_n1(:,3), 'mo', h, omega_n1(:,4), 'ro', ...
    h, omega_n1(:,5), 'bo')
hold on, grid on
% Frequencies from EVP with Timoshenko or EB elements...:
plot(h, wn(1,:), 'cx', h, wn(2,:), 'gx', ...
    h, wn(3,:), 'mx', h, wn(4,:), 'rx', ...
    h, wn(5,:), 'bx')
% hold on, grid on
% Frequencies from Abaqus models:
plot(h, omega_1_1m, 'c*', h, omega_1_2m, 'g*', h, omega_1_3m, 'm*', ...
   h, omega_1_4m, 'r*', h, omega_1_5m, 'b*');
% Plot configurations and layout figure 3:
% a, EB) analytical undamped frequencies from EB-based formula.
% EVP, NxEB/T) freqs from solving EVP in matlab, EB or T elements
% A) Abaqus damped frequencies, B33 (EB) or B31 (T) elements.
legend('1m beams, a/EB', '2m beams, a/EB', ...
    '3m beams, a/EB', '4m beams, a/EB', ...
    '5m beams, a/EB', ...
    '1m beams, EVP/5xT', '2m beams, EVP/5xT', ...
    '3m beams, EVP/5xT', '4m beams, EVP/5xT', ...
    '5m beams, EVP/5xT', ...
    '1m beams, A/T', '2m beams, A/T', ...
    '3m beams, A/T', '4m beams, A/T', ...
```

```
    '5m beams, A/T')
% Change legend Position: Southeast or Eastoutside
lgd = legend; lgd.FontSize = 10; lgd.Location = 'Eastoutside';
% lgd = legend; lgd.FontSize = 11; lgd.Position = [0.18,0.55,0.4,0.35];
axis([0 1050 0 700]) %axis limits, y-axis circular freq
set(gca, 'XTick', [0:50:1050], 'XTickLabelRotation', 90);
% title(sprintf(['Abaqus models "%s":\n' ...
%     'Frequency 1st mode for respective beam heights.\n'...
%     'Analytical vs numerical (NB analyt vals are undamped).'], Name), ...
%     'FontSize', 12)
xlabel('Beam height, h [mm]', 'FontSize', 12)
ylabel('Circular frequency, \omega [rad/s]', 'FontSize', 12)
saveas(gcf, sprintf('%s_analytVSnum_freqs_fig3.png', Name))
end %if
```

Appendix C: Example MATLAB script for MAC-calculation

```
%MAC COMPARISON OF MODE SHAPES FROM TOP SURFACE OF 3D SOLID (VOLUME)
%ELEMENT BEAMS.

% in this example:
% C3D20Rv-5x10x80-5mh300 vs C3D20Rv-5x10x80-5mh950
% beam A: length 5m, height 300mm
% beam B: length 5m, height 950mm

% Change height by changing the name of the loaded files under 1A and 1B.
% Change extracted dofs by changing value of phi_indices(1,i) in loop
clc, clear all, close all
%-------------------------------------------------------------------------

%% 1A
%LOADING THE VARIABLE CONTAINING ALL OUTPUT VARIABLES FROM ABAQUS MODEL A:
%NB: must change path from which Model1_export.mat-file is loaded
% extract the .mat-file from correct folder:
load C:\Users\AndreasH\Documents\Masteroppgave\Abaqus_modeller\ ...
    MAC_ANALYSES\C3D20Rv_5x10x80_5mh300\Model1_export.mat
format long %more decimals

%% 2A
% FINDING THE INDEXES CORRESPONDING TO WANTED NODES

% 2nd column in nodecoord correspond to vertical coord (axis 2/ z-axis)
% TOP PLANE AT z = 200 FOR h300 <-- must change according to h
indices_A = find(nodecoord(:,2) == 200); %Node index

%% 3A
% EXTRACTING THE MODE SHAPES/ EIGENVECTORS FOR THE WANTED NODES, I.E.
% AT THE SPECIFIED INDEXES IN phi and phi_conj

NodesA = length(indices_A); %number of nodes (should be 1371)
phi_indices_A = zeros(1,NodesA); %preallocate vector for speed
RealParts_A = zeros(NodesA,1); %preallocate vector for speed
ComplexParts_A = zeros(NodesA,1); %preallocate vector for speed

for i = 1:NodesA
    % EXTRACTING REAL MODES (REAL PART OF EIGENVECTOR) FROM MODEL A
    % #modes = #columns in mode shape matrix
    phi_indices_A(1,i) = (indices_A(i)-1)*6 + 1; % "+i" --> dof i = Ui
    RealParts_A(i,1) = phi(phi_indices_A(i), 1);%choose col in phi
    % EXTRACTING COMPLEX PART OF MODES FROM MODEL A
    % #modes = #columns in mode shape matrix
    ComplexParts_A(i,1) = phi_conj(phi_indices_A(i), 1);%choose col in phi
end %for
% phi_indices_A;
% REAL MODE SHAPE(S)/ EIGENVECTOR(S) MODEL A:
PSI_A = RealParts_A;
% ASSEMBLING COMPLEX MODES SHAPE(S)/ EIGENVECTOR(S) MODEL A:
complexEigenVec_A = RealParts_A + ComplexParts_A*i; % "a+b*i"
cPSI_A = complexEigenVec_A; %complex mode shapes model A
%-------------------------------------------------------------------------
%-------------------------------------------------------------------------
%% 1B
%LOADING THE VARIABLE CONTAINING ALL OUTPUT VARIABLES FROM ABAQUS MODEL B:
%NB: must change path from which Model1_export.mat-file is loaded
```

```
%extract .mat-file from correct folder:
load C:\Users\AndreasH\Documents\Masteroppgave\Abaqus_modeller\ ...
    MAC_ANALYSES\C3D20Rv_5x10x80_5mh950\Model1_export.mat
format long %want all decimals

%% 2B
% FINDING THE INDEXES CORRESPONDING TO WANTED NODES
% 2nd column in nodecoord correspond to vertical coord (axis 2/ z-axis)
% TOP PLANE AT z = 850 FOR h950 <-- must change according to h
indices_B = find(nodecoord(:,2) == 850); %Node index

%% 3B
% EXTRACTING THE MODE SHAPES/ EIGENVECTORS FOR THE WANTED NODES, I.E.
% AT THE SPECIFIED INDEXES IN phi and phi_conj
NodesB = length(indices_B); %number of nodes (should be 1371)
phi_indices_B = zeros(1,NodesB); %preallocate vector for speed
% phiconj_indices = zeros(1,NodesA); %preallocate vector for speed
RealParts_B = zeros(NodesB,1); %preallocate vector for speed
ComplexParts_B = zeros(NodesB,1); %preallocate vector for speed

for i = 1:NodesB
    % EXTRACTING REAL MODES (REAL PART OF EIGENVECTOR) FROM MODEL B
    % #modes = #columns in mode shape matrix
    phi_indices_B(1,i) = (indices_B(i)-1)*6 + 1; % "+i" --> dof i = Ui
    RealParts_B(i,1) = phi(phi_indices_B(i), 1); %choose col in phi
    % EXTRACTING COMPLEX PART OF MODES FROM MODEL B
    % #modes = #columns in mode shape matrix
    ComplexParts_B(i,1) = phi_conj(phi_indices_B(i), 1);%choose col in phi
end %for
% phi_indices_B;
% REAL MODE SHAPE(S)/ EIGENVECTOR(S) MODEL B:
PSI_B = RealParts_B;
% ASSEMBLING COMPLEX MODES SHAPE(S)/ EIGENVECTOR(S) MODEL B:
complexEigenVec_B = RealParts_B + ComplexParts_B*i; % "a+b*i"
cPSI_B = complexEigenVec_B; %complex mode shapes model B
%--------------------------------------------------------------------------

%% 4
% CONTROL CHECK:
if length(indices_A) ~= length(indices_B)
    disp('Error: The # of indices are not the same for the two models')
    if isequal(indices_A, indices_B) == 0
    disp('And indices for model A and B do not correspond to each other')
    end
elseif isequal(indices_A, indices_B) == 0
    disp('note: Indices for model A and B do not correspond to each other')
end %if

%% 5
% CALCULATE THE MAC-VALUE for mode shapes from model A and B

%MAC calc of real mode shapes
MAC = (abs(PSI_A'*PSI_B)^2)/((PSI_A'*PSI_A)*(PSI_B'*PSI_B))
%MAC calc of complex mode shapes (note: here complex contributions are 0,
%so cMAC should be identical to MAC...):
cMAC=(abs(cPSI_A'*cPSI_B)^2)/((cPSI_A'*cPSI_A)*(cPSI_B'*cPSI_B))
cMACreal=real((abs(cPSI_A'*cPSI_B)^2)/((cPSI_A'*cPSI_A)*(cPSI_B'*cPSI_B)))
%Note that apostrophe ' is the complex conjugate transpose in matlab!
%--------------------------------
```

Appendix D: Example MATLAB script for extraction of output system matrices from Abaqus and solving the EVP to find the eigenfrequencies based on these

```
%% READING IN OUTPUT-MATRICES FROM ABAQUS AND TRANSCRIBING TO FULL FORMAT
clc, clear all, close all
format long %for more decimals
format bank %convert number display from scientific to "normal"

%NB! JOB NAMES MAY NOT CONTAIN "-", ONLY "_".
%For B21-models with 2 elmenents #nodes = 3 --> outAbaqus matrices = 9x9

%% STIFFNESS MATRIX
load Job_B21otrop_consistM_outMatrices_STIF3.mtx
K_outAbaqus = Job_B21otrop_consistM_outMatrices_STIF3; %"sparse" format
% Checks ---------------------
big1 = max(K_outAbaqus(:,1));
big2 = max(K_outAbaqus(:,2));
num_nodes = big1/3;
num_elements = num_nodes - 1;
% -----------------------------
[rows_K, cols_K] = size(K_outAbaqus);

K_Ab = zeros(big1); %preallocate
% Assembling/ transcribing the system stiffness matrix
for i = 1:rows_K
    K_Ab(K_outAbaqus(i,1),K_outAbaqus(i,2)) = K_outAbaqus(i,3);
end %for
% K_Ab;
%Remove rows and columns related to the fixed dofs in the model, i.e.
%rows/cols 1,2 and 8, and additonally axial dofs r/c 4 and 7.
%Remaining kinematic dofs are [r3 r5 r6 r9]'
delrcvec = [1 2 8 4 7]; %incl/excl ax dofs by add/remove [... 4 7]
K_Ab(delrcvec, :) = [];
K_Ab(:, delrcvec) = [];
K_Ab
save Abaqus_K_4dof_otrop_consistM.txt K_Ab -ascii
%-------------------------------------------------------------------------

%% MASS MATRIX
load Job_B21otrop_consistM_outMatrices_MASS3.mtx
M_outAbaqus = Job_B21otrop_consistM_outMatrices_MASS3; %"sparse" format
% Checks ---------------------
big1M = max(M_outAbaqus(:,1));
big2M = max(M_outAbaqus(:,2));
num_nodes = big1M/3;
num_elements = num_nodes - 1;
% -----------------------------
[rows_M, cols_M] = size(M_outAbaqus);

M_Ab = zeros(big1); %preallocate
% Assembling/ transcribing the system mass matrix
for i = 1:rows_M
    M_Ab(M_outAbaqus(i,1),M_outAbaqus(i,2)) = M_outAbaqus(i,3);
end %for
M_Ab(delrcvec, :) = [];
M_Ab(:, delrcvec) = [];
M_Ab
save Abaqus_M_4dof_otrop_consistM.txt M_Ab -ascii
%-------------------------------------------------------------------------
```

```
%% DAMPING MATRIX
load Job_B21otrop_consistM_outMatrices_DMPV3.mtx
C_outAbaqus = Job_B21otrop_consistM_outMatrices_DMPV3; %"sparse" format
% Checks ---------------------
big1C = max(C_outAbaqus(:,1));
big2C = max(C_outAbaqus(:,2));
num_nodes = big1C/3;
num_elements = num_nodes - 1;
% -----------------------------
[rows_C, cols_C] = size(C_outAbaqus);

C_Ab = zeros(big1); %preallocate
% Assembling/ transcribing the system damping matrix
for i = 1:rows_C
    C_Ab(C_outAbaqus(i,1), C_outAbaqus(i,2)) = C_outAbaqus(i,3);
end %for
C_Ab(delrcvec, :) = [];
C_Ab(:, delrcvec) = [];
C_Ab;
save Abaqus_C_4dof_otrop_consistM.txt C_Ab -ascii
%--------------------------------------------------------------------------

%% ANALYTICALLY SOLVING EVP WITH ABAQUS OUTPUT MATRICES:
[eigVec, eigVal] = eig(K_Ab,M_Ab);
PHI = eigVec; %mode shapes (vibration modes)
wn2 = diag(eigVal);
wn = sqrt(wn2); %Natural (circular) frequencies of the system [rad/s]
fn = wn/(2*pi); %Natural frequencies of the system [Hz]
firstNatFreq = fn(1); secNatFreq = fn(2);
% END

% EXTRA: calculation of modal matrices:
% %orthogonalizing/ generalizing the system (decoupling the system) into
% %the modal stiffness- and modal mass matrices:
%
% %MODAL MATRICES based on outMatrices from Abaqus
% KS_Ab = PHI'*K_Ab*PHI; %Modal stiffnes matrix
% MS_Ab = PHI'*M_Ab*PHI; % Modal mass matrix
%
% % to get modal (system) damping:
% %Rayleigh coefficients:
% alpha = 1.13; beta = 7.47e-5;
% %MODAL SYSTEM DAMPING MATRIX based on output matrices from Abaqus
% CS_Ab = alpha*MS_Ab + beta*KS_Ab;
```

Appendix E: Horizontal vibration modes from iSight

Euler-Bernoulli beam element models – B33

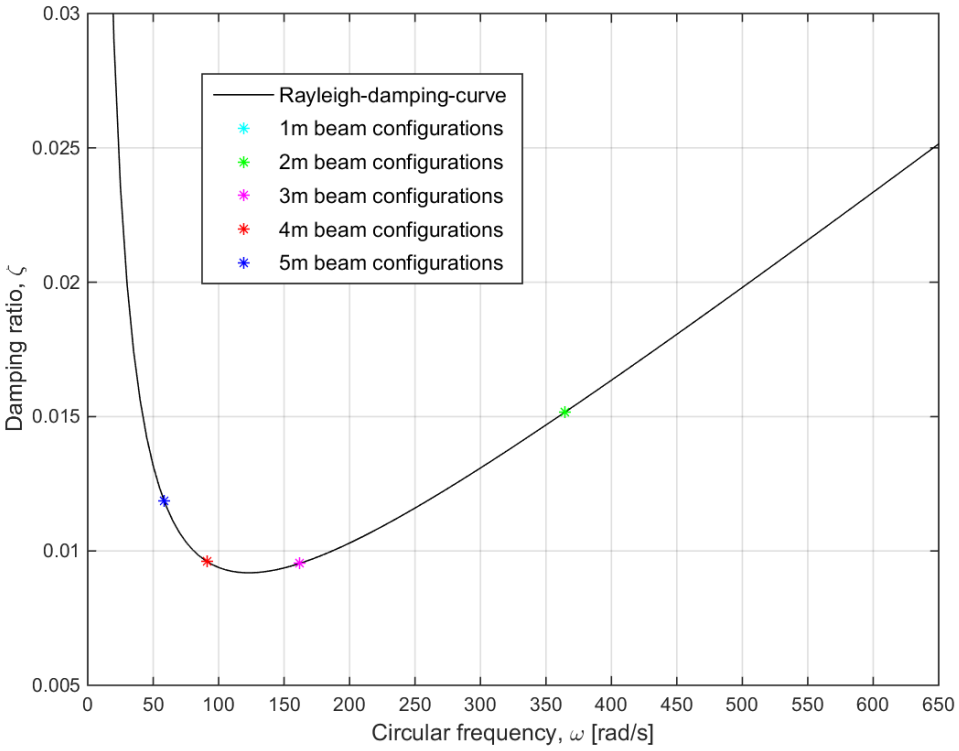

Figure E. 1: Euler-Bernoulli beam model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

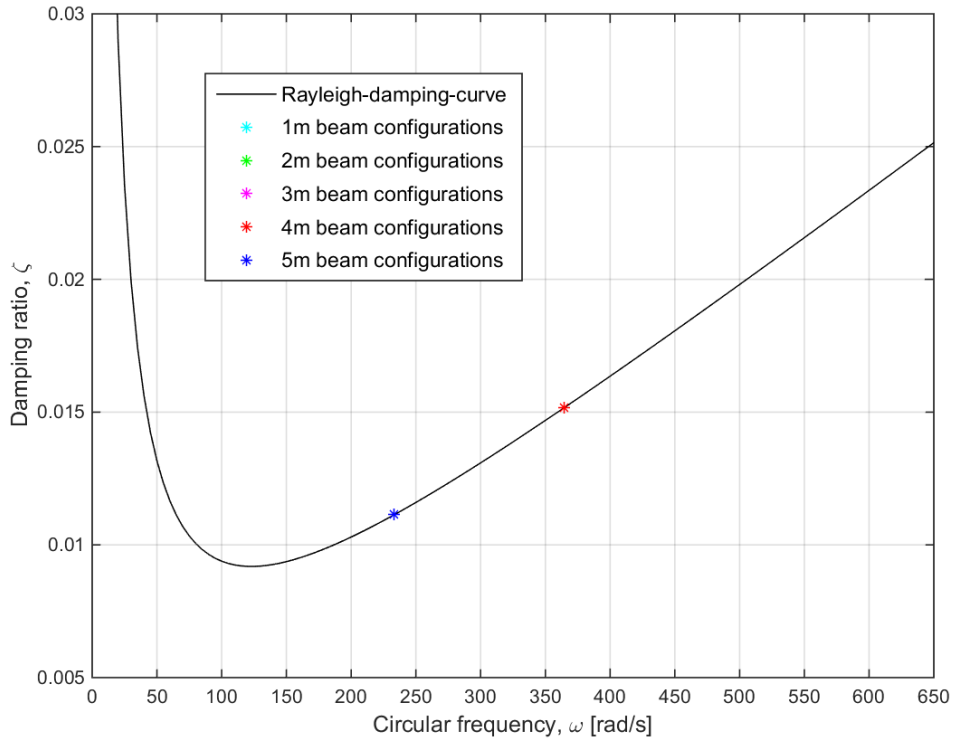

Figure E. 2: Euler-Bernoulli beam model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

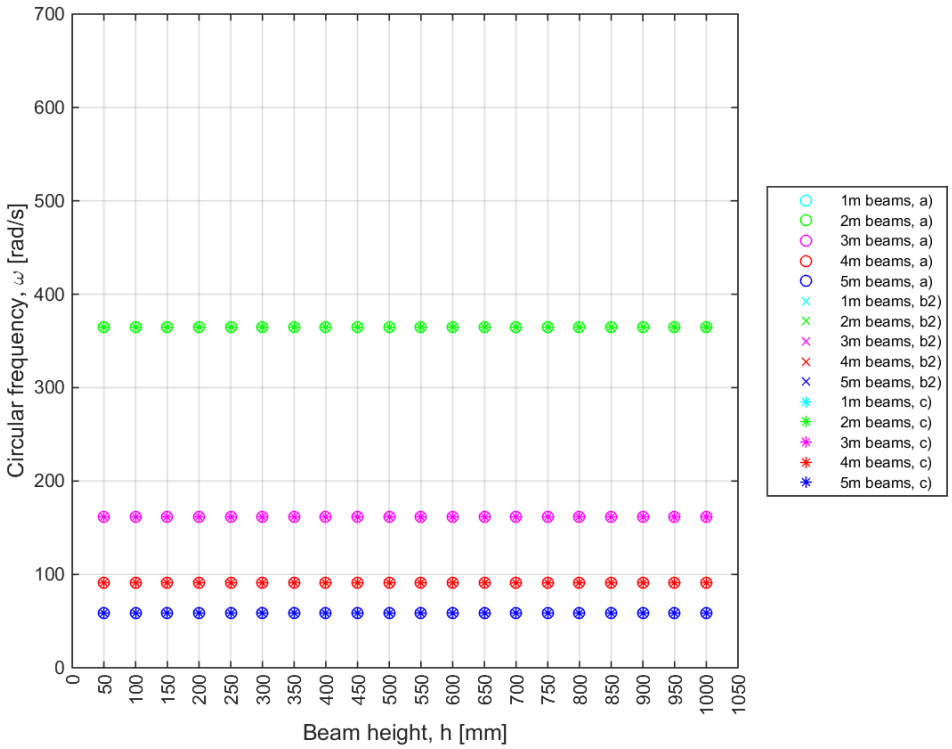

Figure E. 3: Euler-Bernoulli beam model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

Timoshenko beam element models – B31

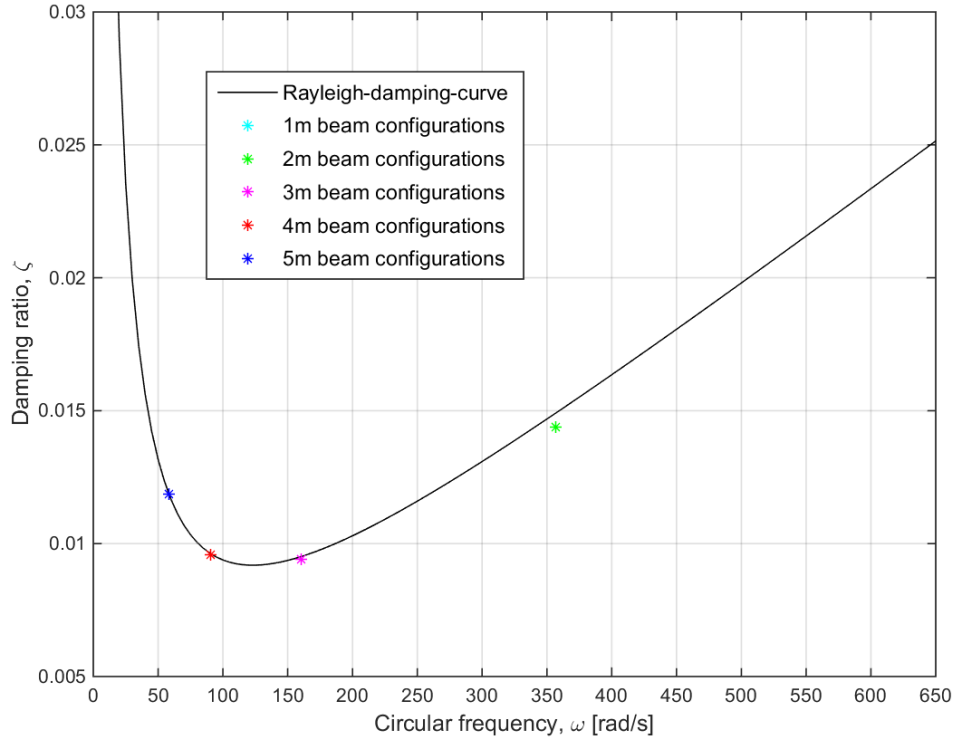

Figure E. 4 Timoshenko beam model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

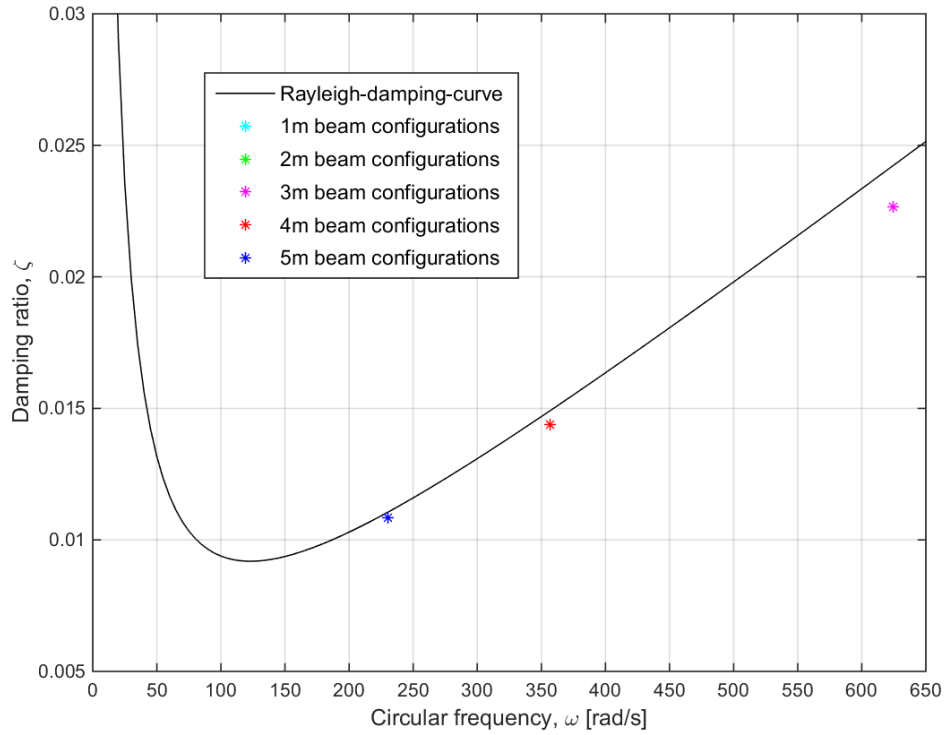

Figure E. 5: Timoshenko beam model. Damping from Abaqus plotted with the Rayleigh-curve. Second horizontal mode of the beam.

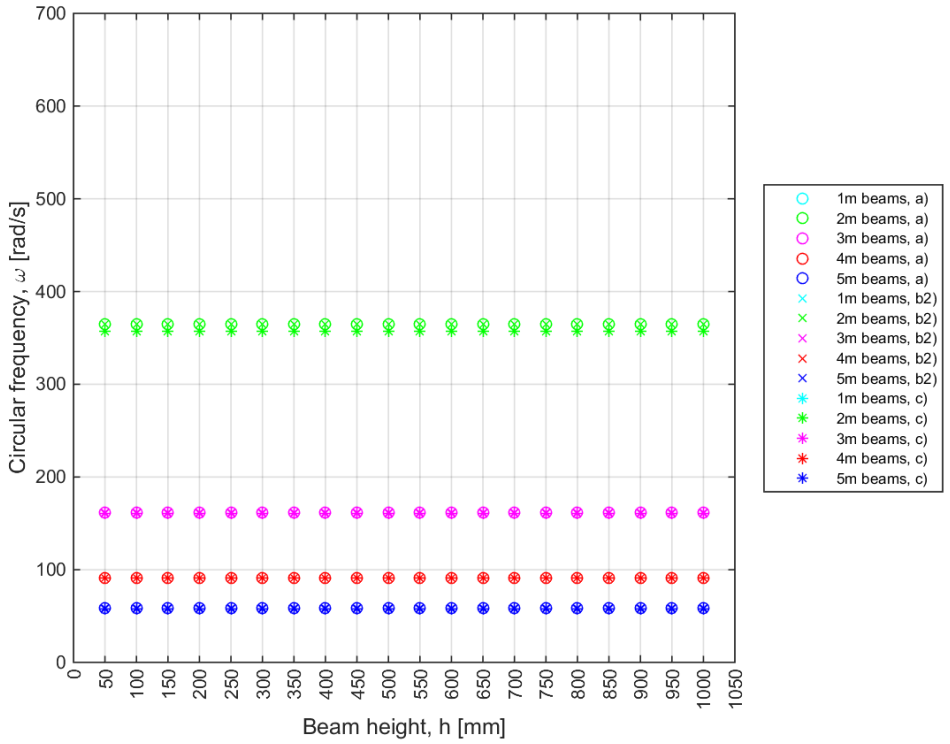

Figure E. 6: Timoshenko beam model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

Conventional shell element models – S8R

Shell thickness as width

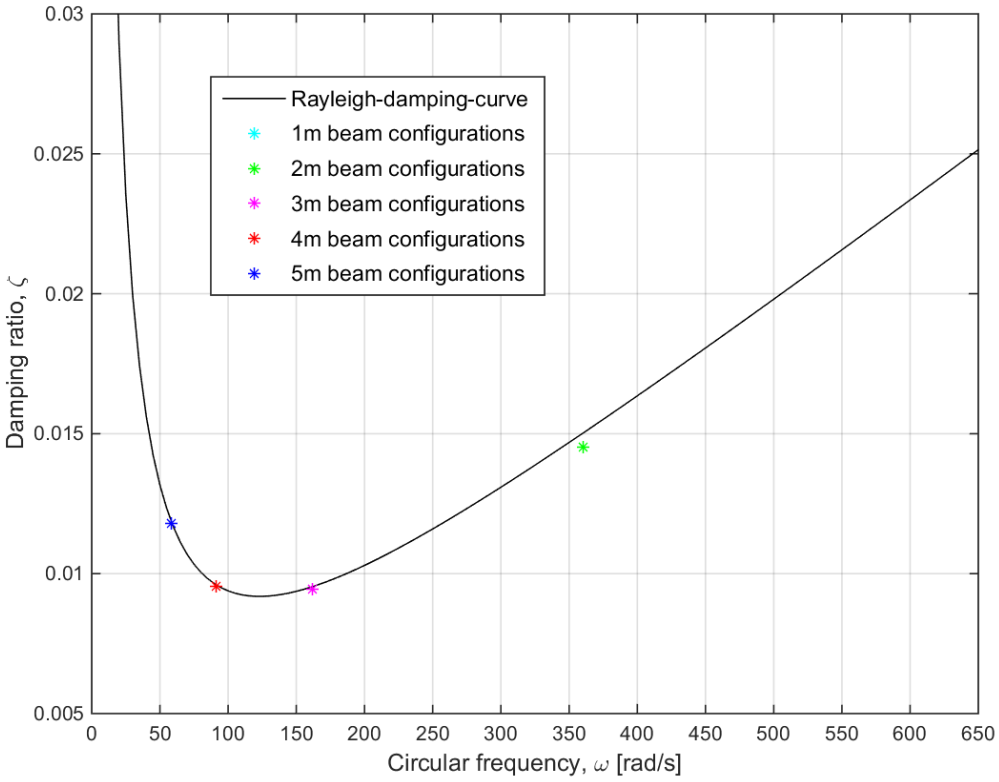

Figure E. 7: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

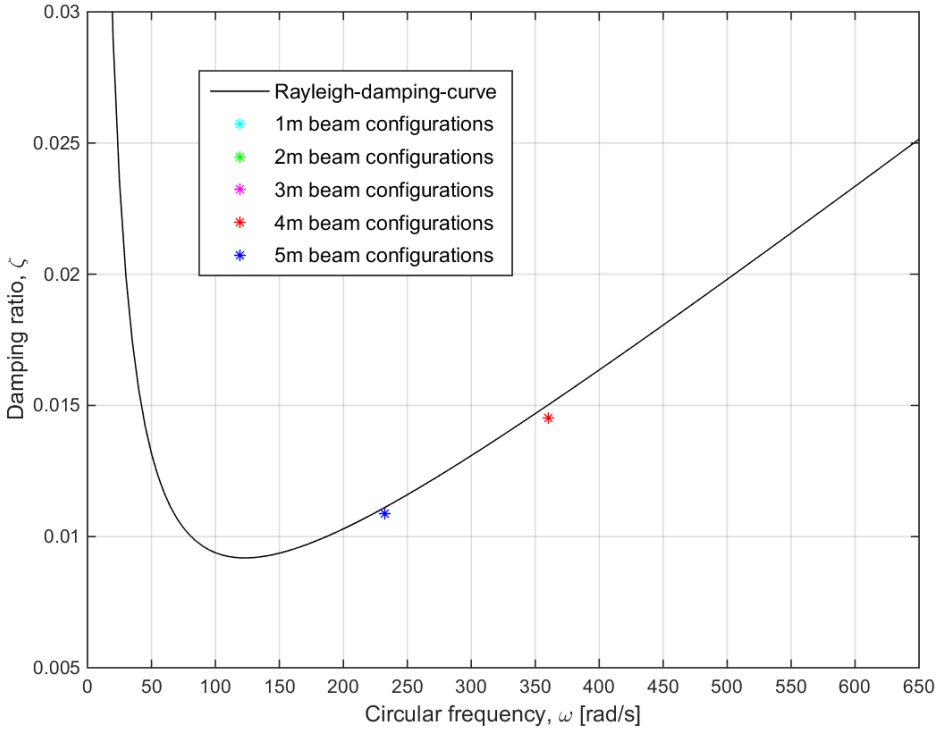

Figure E. 8: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. Second horizontal mode of the beam

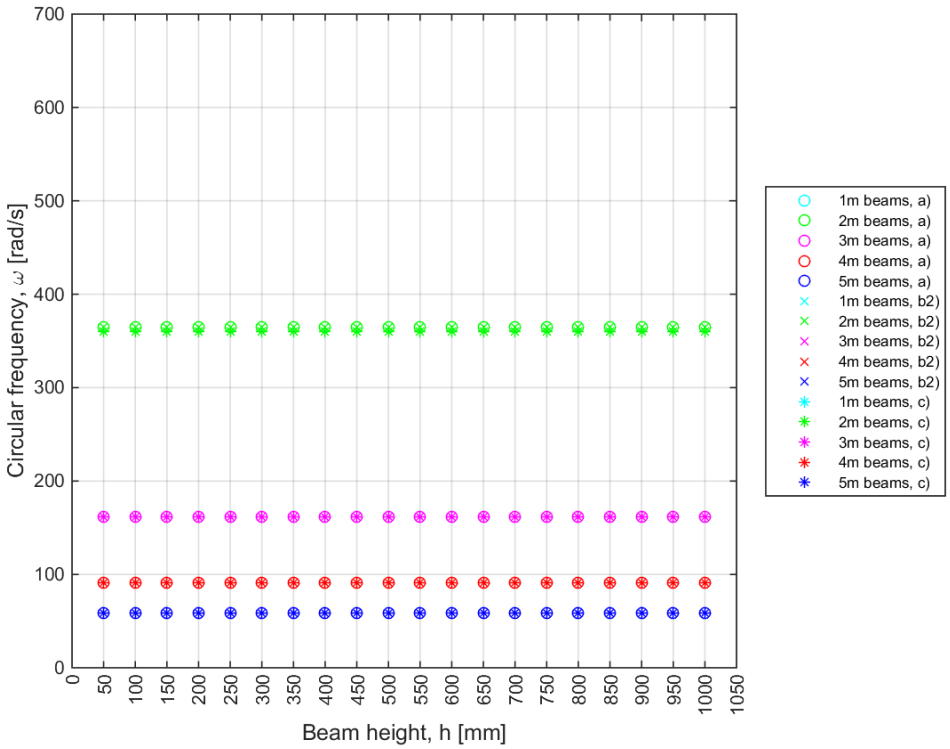

Figure E. 9: Shell model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

Shell thickness as height

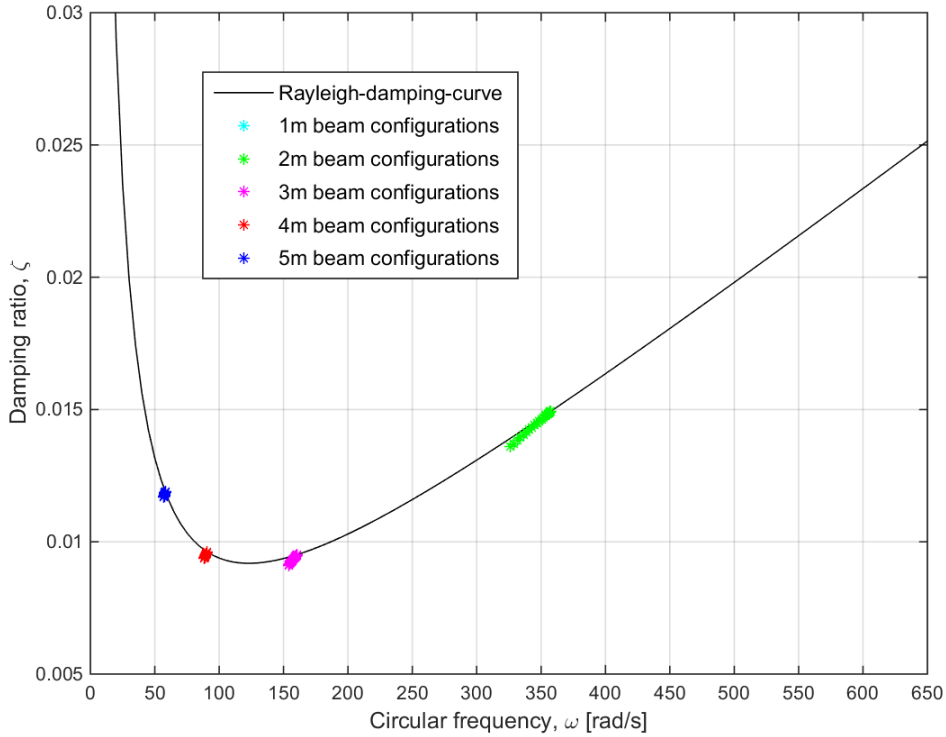

Figure E. 10: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

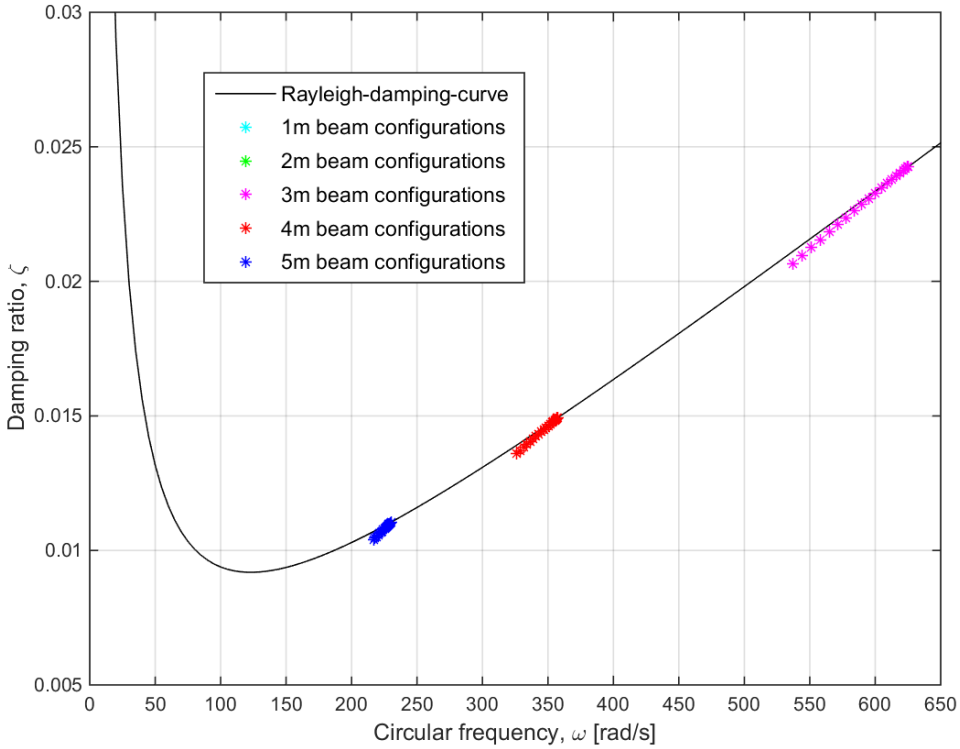

Figure E. 11: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. Second horizontal mode of the beam.

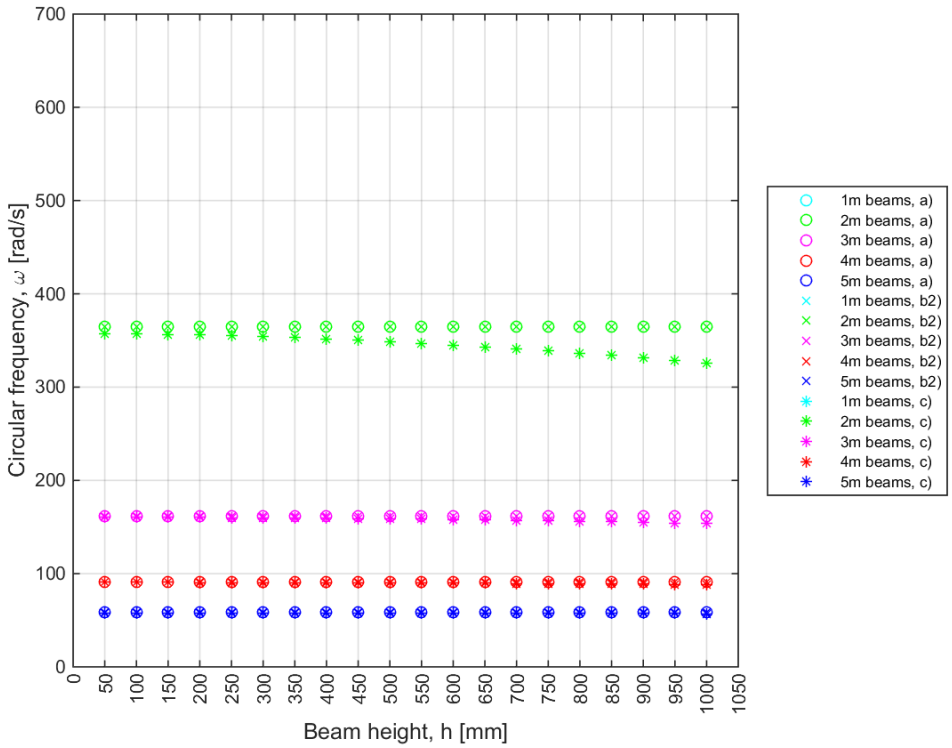

Figure E. 12: Shell model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

3D solid element models – C3D20R

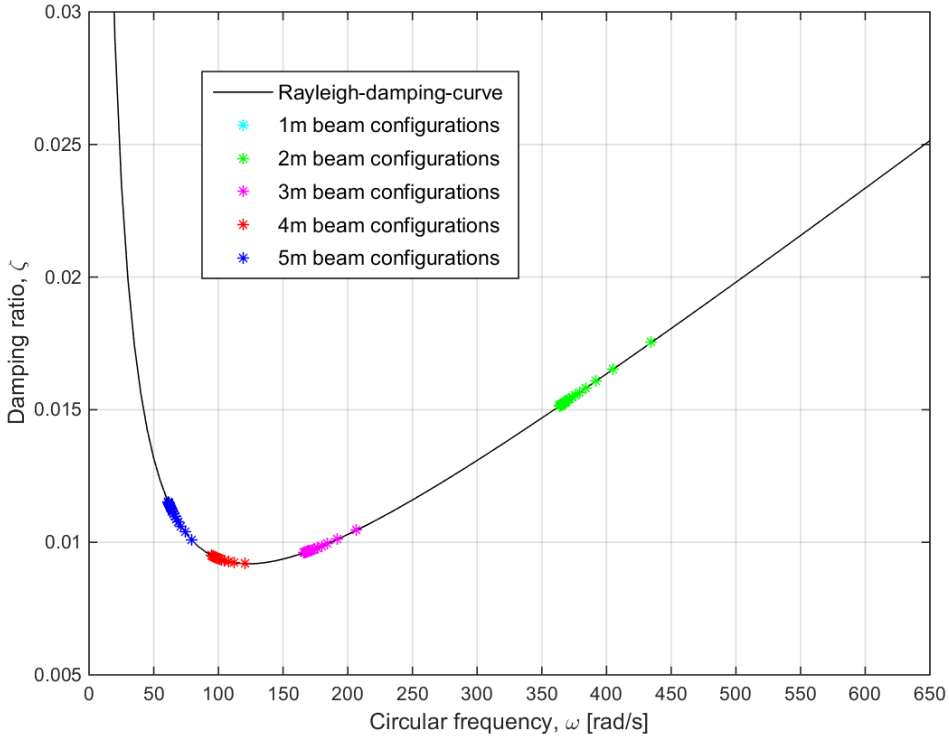

Figure E. 14: 3D solid model. Damping from Abaqus plotted with the Rayleigh-curve. First horizontal mode of the beam.

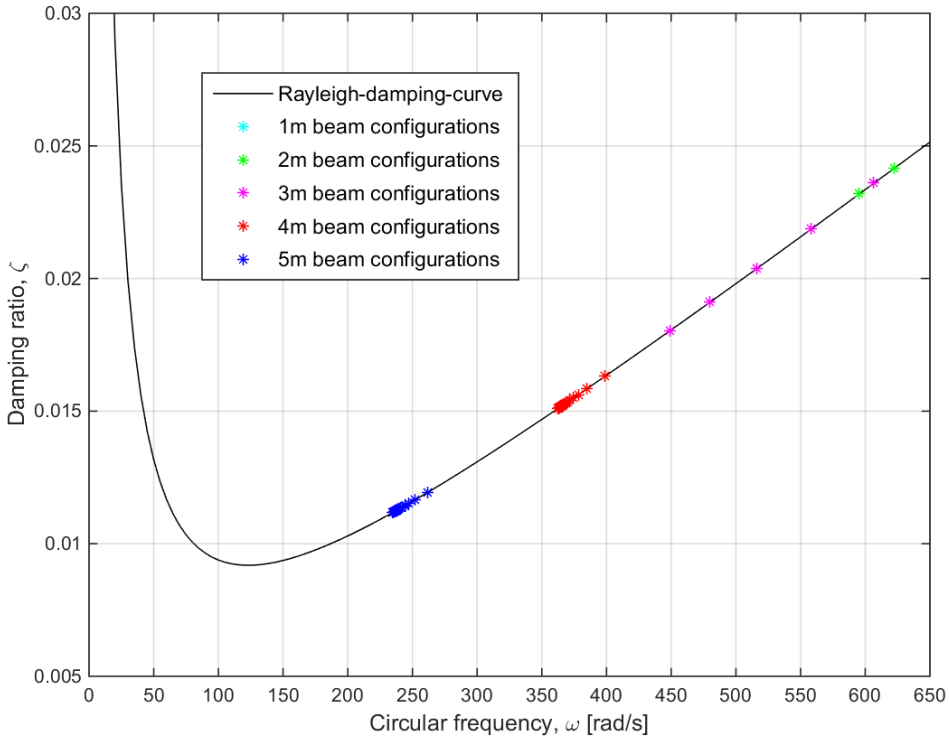

Figure E. 13: 3D solid model. Damping from Abaqus plotted with the Rayleigh-curve. Second horizontal mode of the beam.

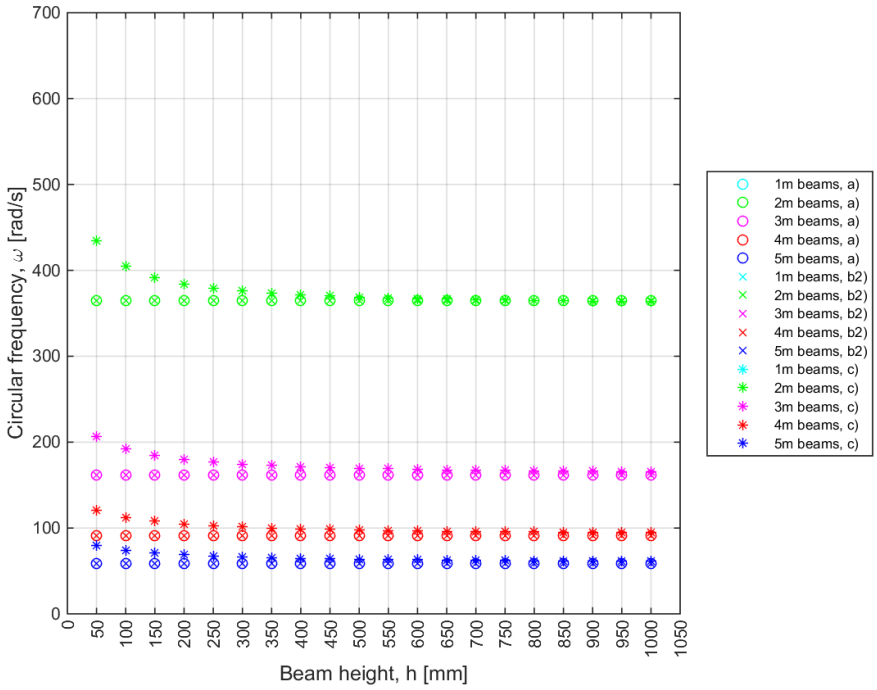

Figure E. 15: 3D solid model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

Appendix F: Vertical vibration modes for shell models with shell thickness as height

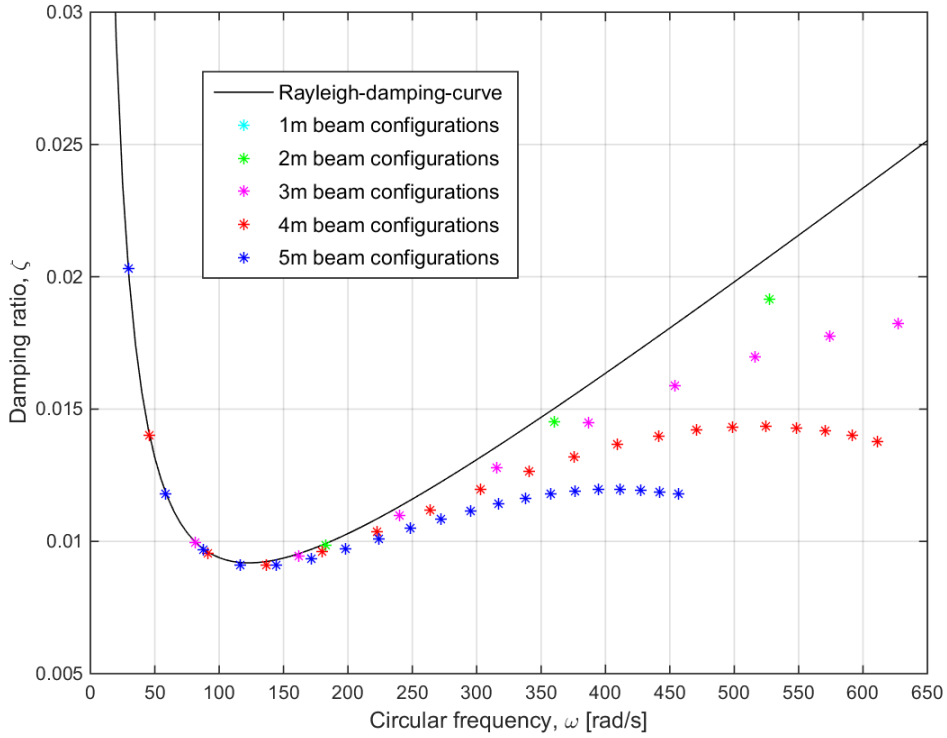

Figure F. 2: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. First vertical mode of the beam.

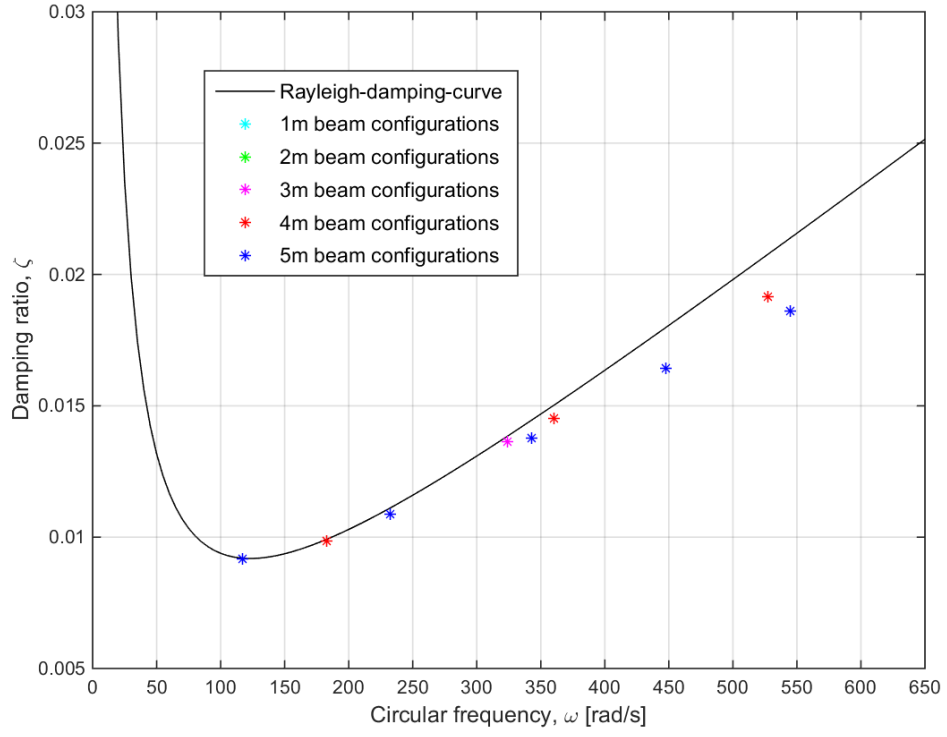

Figure F. 1: Shell model. Damping from Abaqus plotted with the Rayleigh-curve. Second vertical mode of the beam.

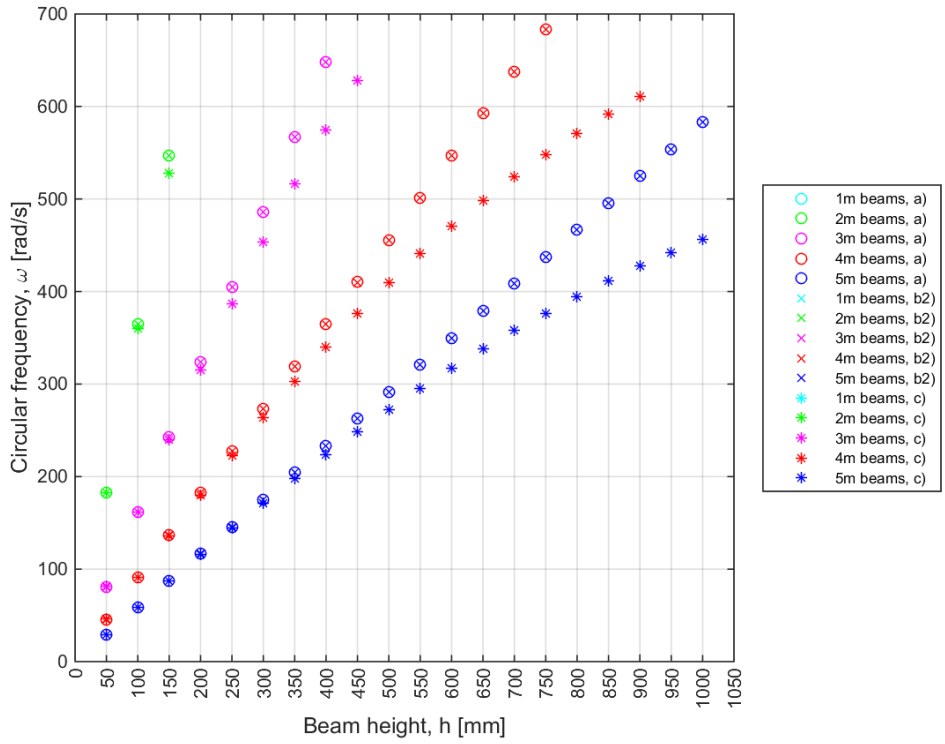

Figure F. 3: Shell model. Eigenfrequencies for different beams are plotted from three different cases. a) Undamped analytical, b1) Damped by 2% analytical and c) Complex frequencies from Abaqus

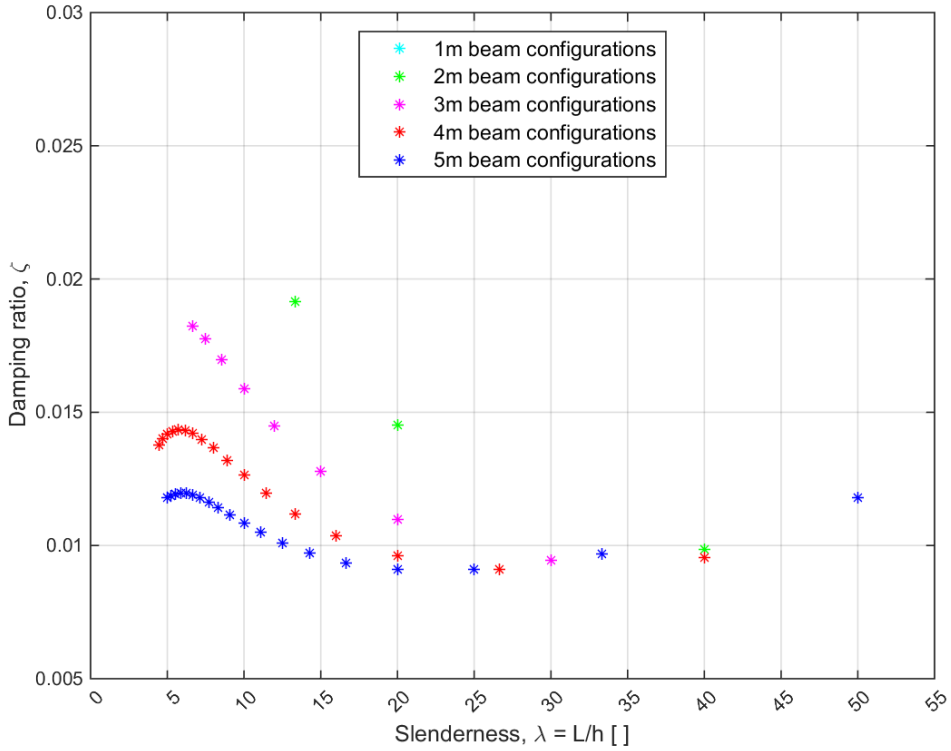

Figure F. 4: Shell model. Damping from Abaqus plotted for the slenderness. L = span length, h = beam height.

Appendix G: Derivation of Rayleigh damping specified on element (material) level and on global system level

Rayleigh damping specified on global system level, means specifying the system damping matrix directly as proportional to both the system mass matrix, **M**, and system stiffness matrix, **K**, so that

$$\mathbf{C_{global}} = \alpha\mathbf{M} + \beta\mathbf{K}$$

where α and β are the Rayleigh damping coefficients, i.e. the proportionality constants, $\mathbf{M} = \sum_{\mathbf{i}} \mathbf{a_i^T m_i a_i}$ and $\mathbf{K} = \sum_{\mathbf{i}} \mathbf{a_i^T k_i a_i}$.

Performing a modal transformation on the system damping matrix (with the undamped eigenvectors (mode shapes) of the system) expresses the modal damping matrix, $\mathbf{C}^*$ as:

$$\mathbf{C^*_{global}} = \alpha\mathbf{M}^* + \beta^*\mathbf{K} = \alpha\boldsymbol{\phi}^\mathbf{T}\mathbf{M}\boldsymbol{\phi} + \beta\boldsymbol{\phi}^\mathbf{T}\mathbf{K}\boldsymbol{\phi}$$

Now, defining Rayleigh damping on element level instead, the element damping matrix reads:

$\mathbf{c_i} = \alpha_i\mathbf{m_i} + \beta_i\mathbf{k_i} \Rightarrow \begin{Bmatrix} \mathbf{c_1} = \alpha_1\mathbf{m_1} + \beta_1\mathbf{k_1} \\ \mathbf{c_2} = \alpha_2\mathbf{m_2} + \beta_2\mathbf{k_2} \\ \cdots \\ etc. \end{Bmatrix}$,where the Rayleigh coefficients may be specified differently for elements that represent different materials.

If the same material is specified for all elements, this leads to $\alpha_i = \alpha$ and $\beta_i = \beta$. The global system contribution from each element "i" may then be written

$$\mathbf{C_i} = \alpha\mathbf{a_i^T m_i a_i} + \beta\mathbf{a_i^T k_i a_i} \Rightarrow \begin{Bmatrix} \mathbf{C_1} = \alpha\mathbf{M_1} + \beta\mathbf{K_1} \\ \mathbf{C_2} = \alpha\mathbf{M_2} + \beta\mathbf{K_2} \\ \cdots \\ etc. \end{Bmatrix} \Rightarrow \mathbf{C} = \sum_{\mathbf{i=1}}^{\mathbf{\#el}} \mathbf{C}_i$$

Performing the modal transformation (not necessary for proving the relation between element and global damping for only one material, but done for completeness) with the undamped mode shapes leads to the following:

$$\mathbf{C}^* = \boldsymbol{\phi}^\mathbf{T}\mathbf{C}\,\boldsymbol{\phi} = \boldsymbol{\phi}^\mathbf{T}(\mathbf{C_1} + \mathbf{C_2} + \cdots)\boldsymbol{\phi}$$

$$= \boldsymbol{\phi}^\mathbf{T}\mathbf{C_1}\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}\mathbf{C_2}\boldsymbol{\phi}$$

$$= \boldsymbol{\phi}^\mathbf{T}(\alpha\mathbf{M_1} + \beta\mathbf{K_1})\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}(\alpha\mathbf{M_2} + \beta\mathbf{K_2})\boldsymbol{\phi} + \cdots$$

$$\left(= \boldsymbol{\phi}^\mathbf{T}\left(\alpha\mathbf{a_1^T m_1 a} + \beta\mathbf{a_1^T k_1 a_1}\right)\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}\left(\alpha\mathbf{a_2^T m_2 a_2} + \beta\mathbf{a_2^T k_2 a_2}\right)\boldsymbol{\phi} + \cdots\right)$$

$$= \boldsymbol{\phi}^\mathbf{T}(\alpha\mathbf{M_1})\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}(\beta\mathbf{K_1})\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}(\alpha\mathbf{M_2})\boldsymbol{\phi} + \boldsymbol{\phi}^\mathbf{T}(\beta\mathbf{K_2})\boldsymbol{\phi} + \cdots$$

$$= \alpha\boldsymbol{\phi}^\mathbf{T}(\mathbf{M_1})\boldsymbol{\phi} + \beta\boldsymbol{\phi}^\mathbf{T}(\mathbf{K_1})\boldsymbol{\phi} + \alpha\boldsymbol{\phi}^\mathbf{T}(\mathbf{M_2})\boldsymbol{\phi} + \beta\boldsymbol{\phi}^\mathbf{T}(\mathbf{K_2})\boldsymbol{\phi} + \cdots$$

$$= \alpha\boldsymbol{\phi}^\mathbf{T}(\mathbf{M_1} + \mathbf{M_2} + \cdots)\boldsymbol{\phi} + \beta\boldsymbol{\phi}^\mathbf{T}(\mathbf{K_1} + \mathbf{K_2} + \cdots)\boldsymbol{\phi}$$

$$= \alpha\boldsymbol{\phi}^\mathbf{T}(\mathbf{M})\boldsymbol{\phi} + \beta\boldsymbol{\phi}^\mathbf{T}(\mathbf{K})\boldsymbol{\phi}$$

$$= \boldsymbol{\alpha}\mathbf{M}^{*} + \boldsymbol{\beta}\mathbf{K}^{*}$$

$$= \mathbf{C}^{*}_{\mathbf{global}}$$

which proves that when all elements represent the same material (there is only one material in the finite element model), modal Rayleigh damping is identical whether specified on element level or on global system level.