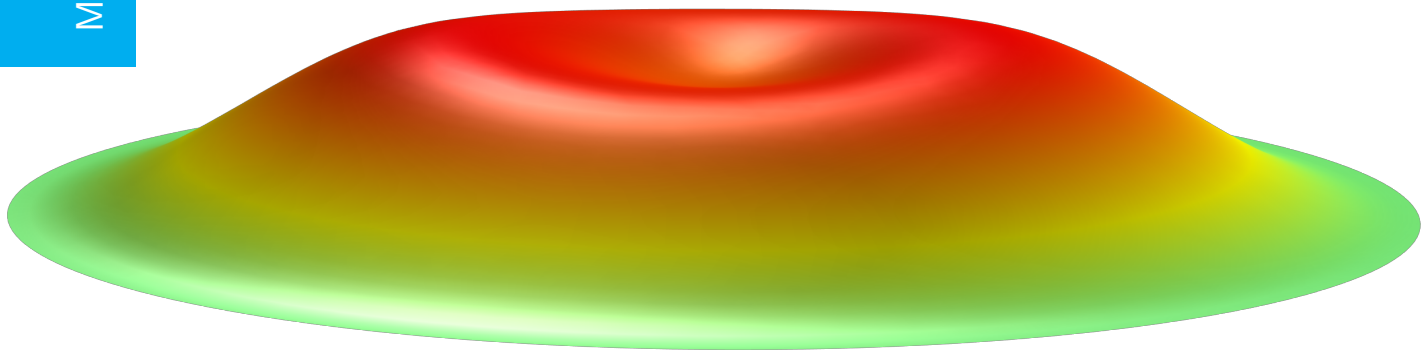
Nonlinear dynamics of graphene membranes

Quality factor limits of membranes caused by mode interaction

V. Bos

Date:	24-1-2020
Supervisor:	P.G. Steeneken
Specialization:	DMN
Student no:	4249135
Report no:	2020.004

Master of Science Thesis



TU Delft Delft University of Technology

Precision and Microsystems Engineering

Nonlinear dynamics of graphene membranes

Quality factor limits of membranes caused by mode interaction

Master of Science Thesis

For the degree of Master of Science in Mechanical Engineering at Delft University of Technology

V. Bos

January 18, 2020

Student no: 4249135
Report no: 2020.004

September 2019 - January 2020

Dynamics of Micro and Nanosystems
Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Table of Contents

Preface

This thesis is about dynamics, a field of physics concerning motion of matter. Quietly, dynamics plays a large role in our daily life, from music we listen to the path of the moon we look at during night. It's quite fascinating that a lot of those dynamic events can be described by mathematics, giving us understanding, the ability to make predictions and the knowledge for new designs, enabling the impossible. Wondering serves as a point of departure for asking questions, driving research to new fields every time. This thesis aims to answer some questions, although in the end, new rose. I hope that during reading this thesis, you will be fascinated about the wonderful world of nonlinear dynamics, and perhaps further discover it.

Acknowledgements

I would like to thank my supervisor prof.dr. P.G. Steeneken for offering me this wonderful project and the many interesting discussions we've had.

I would also like to thank dr.ir. F. Aljani for his help, especially on the numerics, and ir. A. Keşkekler (A.Keskekler-1@tudelft.nl), for involving me in his project and sharing experimental results.

Furthermore, I would like to thank Willian for his nice company while travelling to Delft every day. Last, I would like to thank my fellow students in the office for their fruitful discussions and encouragements.

Delft, University of Technology V. Bos
January 18, 2020

Chapter 1

Introduction

Micro and nano structures open up a world of new possibilities for sensors and actuators. For their performance, efficient energy handling (expressed in the quality factor) is very important. Bad energy management is related to high damping, yielding low sensor sensitivity. Graphene is a new, ultra strong material, but suffers from high damping. The physical mechanism behind this is still unknown. Possibly, nonlinear dynamics play a role in this damping mechanism as it can create 'mode coupling'. To illustrate the interaction between damping and this 'mode coupling', two examples are given in the following.

Imagine two neighbours living in a semi-detached house. The first habitant is very sparing, and buys double-glazed windows to keep all warmth inside. The second habitant is a student, not caring about warmth or durability, leaving open all windows. Despite the efforts of our sparing fellow to prevent heat flow to the environment, he still has to burn gas, as heat flows through the partition-wall to the student's house. The coupling between the two houses limits the maximum energetic efficiency he can get.

In dynamic structures, the same can happen. For example, consider an A-string on a guitar. When it is played gently, you can hear it's fundamental tone, and very softly, the higher harmonics. While listening, the pitch remains steady, as all energy remains in the tones you put it with your finger. Waiting patiently, the sound fades out as energy is lost to the surrounding air. However, when touching the string more boldly, something special may happen. The higher harmonics tap energy from the fundamental A-tone, generating a broader sound. This coupling is due to nonlinear dynamic effects, which arise at large vibration amplitude.

Modelling nonlinear dynamics proves to be very challenging and requires extensive computational resources, or is simply impossible. The only possible approach is to simplify the problem, while preserving the nonlinear character.

Therefore, the value of this work is two-sided. First, methods are presented to obtain nonlinear coupling, which could be used for very different purposes. Second, presented methods are used to investigate how this nonlinear coupling serves as a damping mechanism.

Research questions

From the preceeding problem statement, the research question can be defined as follows:

> *How can the nonlinear dynamics of a structure be described in a simple model using Finite Element Modelling (FEM) and to what extend do those nonlinearities limit the quality factor?*

To answer this, the following sub-questions are formulated:

1. How can a simple mathematical model be built to describe the nonlinear dynamics of a structure using FEM?
   (a) Which approaches exist to model nonlinear dynamics and which one is most suitable for non-analytic, multi-physical models?
   (b) How can the selected method be implemented and improved such that it can be used for 2D structures?
   (c) How can in-plane dynamics be modelled using eigenmodes?
2. How does nonlinear stiffness influence the dynamic behaviour of simple structures?
   (a) What nonlinear dynamic effects can be observed numerically during direct forced vibration of simple structures?
   (b) What nonlinear dynamic effects can be observed during parametric excitation of a string or circular membrane?
   (c) To what extend does the dynamic response of a square or circular membrane resemble experimental results?
   (d) To what extend can Fermi-Pasta-Ulam (FPU) behaviour be observed during free vibration of strings and membranes?
3. How can the energy flow due to nonlinear mode coupling be described?
   (a) How can mode coupling damping be extracted from free vibration simulations?
   (b) How does mode coupling damping in membranes depend on initial conditions like vibration energy and noise temperature?
   (c) How does mode coupling damping in membranes depend on geometric properties like diameter and aspect ratio?

Thesis outline

In chapter 2, the first sub-question is treated. Different methods from literature are discussed, whereafter the most promising is selected. The chapter continues describing the improvement and implementation of the selected method. In chapter 3, the implemented method is used to analyse a couple of simple structures. Several nonlinear dynamic effects are shown, and finally compared to experimental data from literature. Chapter 4 focusses on mode coupling

damping, the final goal of this work. Energy flow is described qualitatively as a function of different parameters, to capture the effect. Results are discussed in chapter 5, leading to the conclusion written in chapter 6. Recommendations regarding future work are mentioned in chapter 7, finalizing this thesis.

Chapter 2

Methods

This chapter describes the methods used to analyse dynamics of (geometric) nonlinear structures. Besides, it serves as a guide for someone who would like to use the developed code himself, or perform a comparable study.

In section 2-1, a brief introduction is given into the modelling of nonlinear dynamics. Section 2-2 continues discussing existing approaches, followed by a detailed explanation of the Stiffness Evaluation Procedure (STEP). Then, section 2-4 treats the difficulty of in-plane mode handling, proposing several solutions. Finally, section 2-5 elaborates on the implementation of the STEP method.

2-1 Introduction

2-1-1 Sources of nonlinearity

Most physical laws are nonlinear. Fluid mechanics describes dozen of nonlinear effects, partly only known by empirical relations. In structural mechanics, several sources can be identified. The kinematic relations (describing the relation between displacements and strains) are quadratic and coupled, and also most damping mechanisms are nonlinear. On top of that, structures can be coupled to nonlinear forces like magnetic or electric, or exhibit fluid-structure interaction.

In structural mechanics, vibration amplitudes remain usually small and nonlinear effects can be neglected (or linearised). However, structures with high aspect ratio can have low bending stiffness in one or more directions, allowing for large displacements. Here, nonlinear effects become important. Typical structures are strings, thin beams, plates and membranes. For structures with both ends fixed, stretching of the midplane causes the out-of-plane stiffness to increase for large displacements (hardening). Furthermore, cantilevers show nonlinear effects related to inertia. Due to large displacements, a nonlinear coupling arises between axial and transversal vibrations, affecting the effective inertia.

This project mainly focusses on static nonlinearities (nonlinear stiffness) which arise from the nonlinear kinematic relations.

2-1-2 Reduced Order Model (ROM)

Predicting the dynamic behaviour of geometric nonlinear structures can be challenging, as it may require huge computationally resources. In order to simplify the problem, a reduced order model can be constructed, including non-linearities. Predicting the behaviour by solving the equations now becomes feasible. To add nonlinearities to a reduced order model, a few methods are available. This work mainly focusses on geometric nonlinear stiffness. However, the methods could also be used for other sources of nonlinearity. In the following chapter, the implementation of a method proposed in literature is described including some improvements. In the appendix an example of the used techniques applied to a simple 2 degree of freedom (dof) structure can be found.

Linear reduced order models

Modelling the dynamics of a structure can computationally be very expensive. Modal analysis allows for splitting up the structure according to it's harmonic behaviour. Each eigenmode describes the vibration at a certain frequency, and according to the spatial and harmonic content of the loadcase a proper selection of eigenmodes can be made. This set of eigenmodes is the basis for the reduced order model, which is computationally way cheaper to solve. After applying the loadcase and initial conditions, the problem can be solved easily. The solution is transformed back to the physical domain to obtain the result. This method is well known and straightforward for linear systems. For systems having nonlinear behaviour that cannot be neglected, the method is still very useful, but has to be extended.

Nonlinear reduced order models

Nonlinear reduced order models include nonlinear terms in the reduced set of equations. Those could be any type of nonlinear functions. However, as most functions can be approximated well by Taylor series for a limited range around a certain point, usually quadratic and cubic terms are introduced. They can describe symmetric and non-symmetric behaviour as well as coupling between modes.

The nonlinear terms in the reduced equations can directly yield from the full set of equations describing the complete system. Another way is to determine the nonlinearities for each eigenmode after the reduced basis is formed. This could be done both numerically or experimentally. In the next section, some methods to construct a nonlinear ROM are discussed.

2-2 Methods for identification of modal nonlinear stiffness

There exist at least 3 methods to identify nonlinear stiffness in the modal domain. For simple structures, analytical methods can be used. Some examples of feasible structures are given in section 2-2-1. For more complicated structures, methods based on FEM are available. Those are discussed in sections 2-2-2 and 2-2-3.

2-2-1 Analytic approach

For structures that can be described by continuous mathematical functions, an analytical approach could be used. The great benefit of this method is that the obtained values can be still a function of structure properties like Youngs's modulus and thickness. This provides insight in the mechanisms behind the obtained values. The results are usually used as a reference for other methods. Usually, some assumptions are made to simplify the equations, such that they become suitable for analytical methods. In literature, some structures are analysed, for example:

- a string (fixed-fixed, 2D, with pre-stress, without bending stiffness [1], [2])
- a beam (simply supported, 2D, nonlinear bending stiffness [3])
- a circular membrane (fixed, 2D axisymmetric, without bending stiffness [4])
- a circular plate (free at the edge, 2D, nonlinear bending stiffness [5])

To apply this method, fundamental knowledge of the mechanics as well as a good understanding of nonlinear differential equations is required. Due to the complexity of the nonlinear partial differential equations, results may look at first sight quite incomprehensible, making it less accessible.

2-2-2 FEM elementwise approach

For more complicated structures, a FEM based method should be applied. K. Markestein proposed a method [6] in which the modal nonlinear stiffness could be obtained from the elemental nonlinear stiffness. This method is fast and precise. However, it should be implemented in the FEM package itself, which has to be accessible. Second, it can only work with the elements available in the package. Third, in-depth knowledge about FEM and the governing physics is required. Fourth, static condensation (on which will be elaborated in section 2-4-2) is not easily applicable.

2-2-3 STEP

Another FEM based method was proposed by Muravyov and Rizzi [7], called Stiffness Evaluation Procedure (STEP). The method constructs modal nonlinear stiffness terms from a set of solutions to static loadcases. Therefore, it requires a FEM package which can handle geometric nonlinearities, but requires no access to the FEM source code.

As the method requires a set of static analysis, the procedure can be regarded slower than Markensteins's method. However, all static problems are independent of each other (they do not require the solution of another problem), so it is perfectly suited for parallel computing. As any FEM package fulfilling the requirements can be used, a broad range of elements can be used. Moreover, also nonlinear stiffness due to other physical effects could be captured. Besides, no in-depth knowledge is required of the physics and discretization. Importantly, in

this thesis it is shown that for 2D structures, static condensation can be easily integrated in STEP (section 2-4-2), which can significantly reduce computational cost.

As the STEP method is regarded very promising, it is extensively discussed in the next section.

2-3 STEP

The STEP method is explained in two papers of Rizzi and Muravyov, applying the method to a fully clamped beam and a plate [7]. Here, it will be briefly explained in a more descriptive way. First, the method is located in the complete modelling process (section 2-3-1). Thereafter, the method is explained and illustrated by some schedules in section 2-3-2. Finally, a mathematical description is given in section 2-3-3.

2-3-1 Modelling overview

Modelling of nonlinear dynamics using the STEP method is summarized in Figure 2-1:

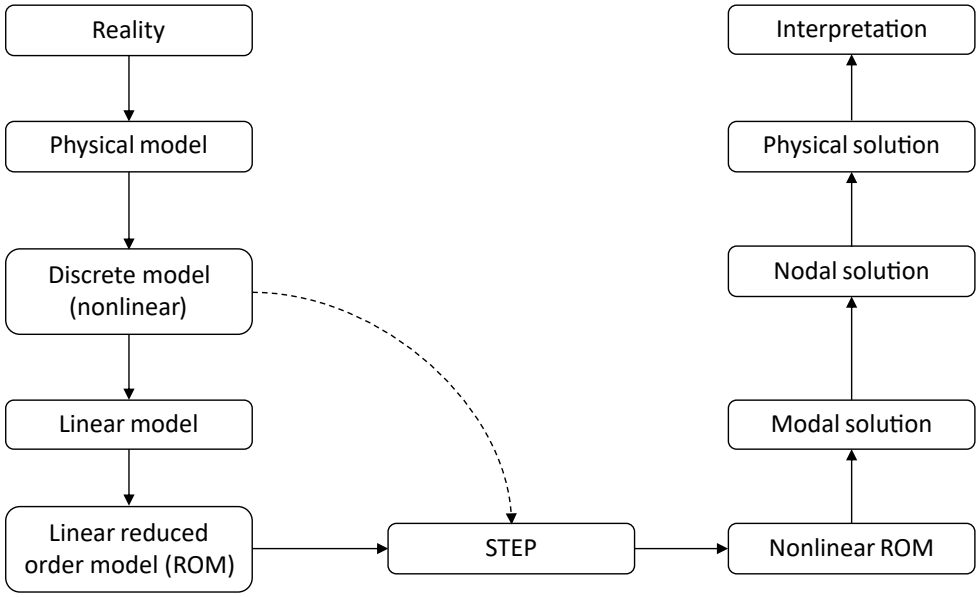


Figure 2-1: Overview modelling nonlinear dynamics using STEP method. The method is located after the construction of a linear ROM. Besides, it uses information coming from the full nonlinear model (indicated as the dotted line). Using this information and the linear ROM, the nonlinear ROM is constructed.

The method pops up after the linear reduced order model is defined, and uses the full nonlinear FEM model to extract the nonlinear modal stiffness. In the following sections, it is explained more detailed.

2-3-2 Procedure

The linear reduced order model consists of a set of decoupled linear equations, each describing the linear dynamics of a mode (figure 2-2, part a). However, in reality each mode also exhibits a nonlinear stiffness (figure 2-2, part b) (This information was lost when the structure was linearised in order to perform a modal analysis.) The nonlinear stiffness can be identified (figure 2-2, part c) by prescribing a displacement Δx and evaluating the needed force F_e (the system can be solved statically as we impose zero velocity and acceleration). Using this result, the nonlinear stiffness coefficient b can be determined.

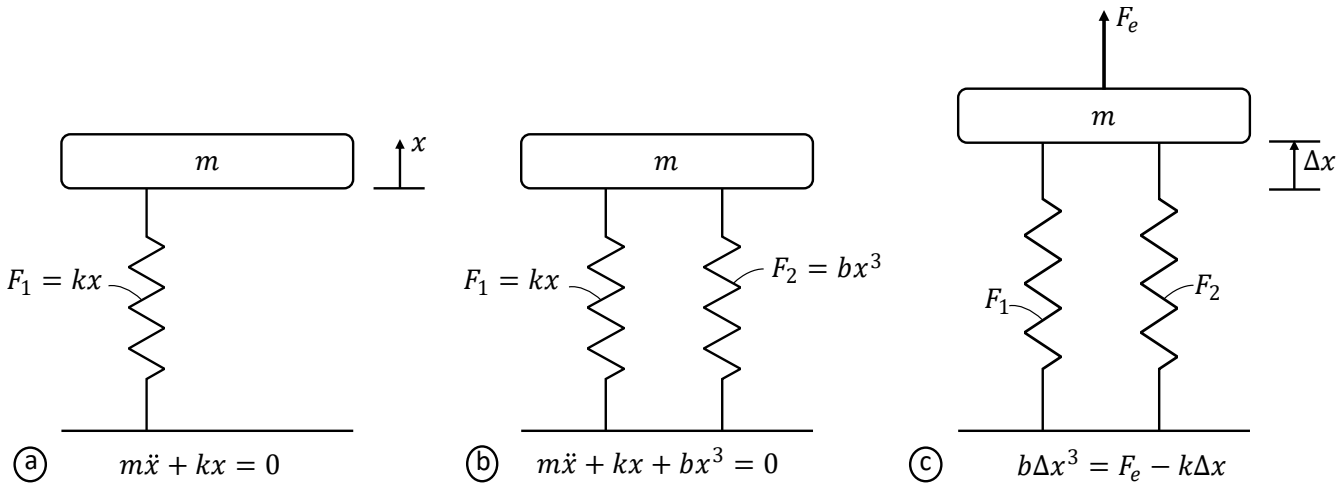


Figure 2-2: STEP method: 1) Linear uncoupled system obtained from modal analysis 2) In reality, nonlinear stiffness is present 3) The nonlinear stiffness can be identified by prescribing a modal displacement and evaluating the modal force

The last part (figure 2-2, part c) can be subdivided into the following actions (shown in figure 2-3):

1. Define a set of displacement fields u. Each field is a linear combination of the linear eigenvectors $\boldsymbol{\phi}$ included in the ROM consisting of L modes. The contribution of each eigenvector is indicated by it's modal amplitude q_j, such that $u_k = \sum_{j=1}^{L} \boldsymbol{\phi_j} q_{jk}$. In Figure 2-3b, this is illustrated for $u = \boldsymbol{\phi_1} q_1$.

2. Subject the full nonlinear model to the set of prescribed displacement fields u and compute nodal reaction forces $\boldsymbol{F}$ (Figure 2-3c)

3. Map nodal reaction forces $\boldsymbol{F}$ to modal forces $\boldsymbol{f}$ (Figure 2-3d)

4. Determine nonlinear stiffness terms (b_1 in the example) (Figure 2-3e)

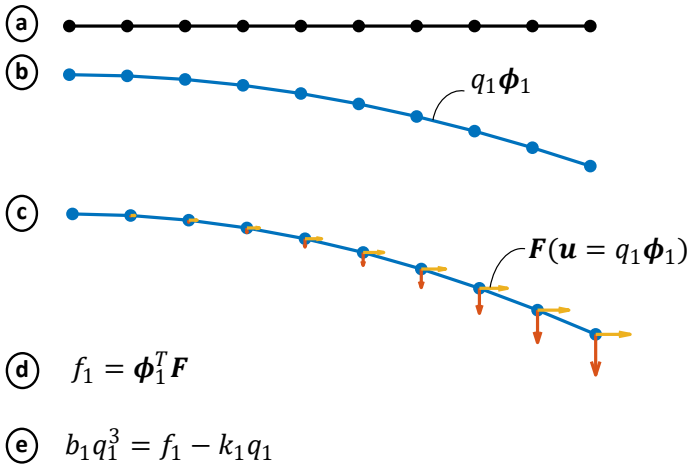


Figure 2-3: STEP method: a) undeformed configuration b) prescribe displacement field c) evaluate nodal reaction forces d) calculate modal force e) solve for nonlinear stiffness coefficient

In this way, all nonlinear stiffness coefficients can be determined, including those coupling the mode to each other. A mathematical description is given in the next section.

2-3-3 Step method: mathematical explanation

The following part is a shortened copy (literal citation) of part of the paper of Muravyov and Rizzi [7], which describes the STEP method, and is reproduced here for convenience of the reader.

> "The equations of motion of a multiple degree-offreedom, viscously damped geometrically nonlinear system can be written in the form:
>
> $$\boldsymbol{M}\ddot{\boldsymbol{X}}(t) + \boldsymbol{C}\dot{\boldsymbol{X}}(t) + \boldsymbol{K}\boldsymbol{X}(t) + \boldsymbol{\Gamma}(\boldsymbol{X}(t)) = \boldsymbol{F}(t) \tag{2-1}$$
>
> where $\boldsymbol{M}, \boldsymbol{C}, \boldsymbol{K}$ are the mass, proportional damping, and linear stiffness matrices, respectively, and $\boldsymbol{X}$ is the displacement response vector and $\boldsymbol{F}$ is the force excitation vector. For the problems of interest, the nonlinear stiffness force vector $\boldsymbol{\Gamma}$ is more than adequately represented by second and third order terms in $\boldsymbol{X}$ and vanishes for small displacements.
>
> A set of coupled modal equations with reduced degrees-of-freedom is first obtained by applying the modal coordinate transformation
>
> $$\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{q} \tag{2-2}$$
>
> to Eq 2-1, where $\boldsymbol{\Phi}$ is the eigenvectors obtained from (2-1) without $\boldsymbol{C}$, $\boldsymbol{q}$ is the vector of modal coordinates, and the time dependence is implied. Generally, a subset of L eigenvectors are included in the solution such that $L < N$, and N is the number of physical degrees of freedom. This coupled set is expressed as

$$\widetilde{\boldsymbol{M}}\ddot{\boldsymbol{q}} + \widetilde{\boldsymbol{C}}\dot{\boldsymbol{q}} + \widetilde{\boldsymbol{K}}\boldsymbol{q} + \gamma\left(q_1, q_2, \ldots, q_L\right) = \widetilde{\boldsymbol{F}} \tag{2-3}$$

$$\begin{aligned}
\widetilde{\boldsymbol{M}} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{\Phi} = [\boldsymbol{I}] && \text{(2-4)}\\
\widetilde{\boldsymbol{C}} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{\Phi} = [2\zeta_r\omega_r] && \text{(2-5)}\\
\widetilde{\boldsymbol{K}} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{K}\boldsymbol{\Phi} = \left[\omega_r^2\right] && \text{(2-6)}\\
\gamma &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Gamma} && \text{(2-7)}\\
\widetilde{\boldsymbol{F}} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{F} && \text{(2-8)}
\end{aligned}$$

where $q_1, q_2, ..., q_L$ are the components of $\boldsymbol{q}$, and ω_r are the undamped natural frequencies. By writing the nonlinear force vector in the form

$$\gamma_r\left(q_1, q_2, \ldots, q_L\right) = \sum_{j=1}^{L}\sum_{k=j}^{L} a_{jk}^r q_j q_k + \sum_{j=1}^{L}\sum_{k=j}^{L}\sum_{l=k}^{L} b_{jkl}^r q_j q_k q_l, \qquad r = 1, 2, \ldots, L \tag{2-9}$$

the problem of determining the nonlinear stiffness is reduced from one in which a large set of simultaneous nonlinear equations must be solved to one involving simple algebraic relations, as will be subsequently shown. Its evaluation entails solving for the coefficients a_{jk} and b_{jkl} using a new procedure.

The procedure is based on the restoration of nodal applied forces by prescribing nodal displacements in both linear and nonlinear static solution settings. The total nodal force $\boldsymbol{F}_{\mathrm{T}}$ may be written in physical coordinates as

$$\boldsymbol{F}_{\mathrm{T}} = \boldsymbol{F}_{\mathrm{L}} + \boldsymbol{F}_{\mathrm{NL}} = \boldsymbol{K}\boldsymbol{X}_{\mathrm{c}} + \boldsymbol{\Gamma}\left(\boldsymbol{X}_{\mathrm{c}}\right) \tag{2-10}$$

where $\boldsymbol{X}_{\mathrm{c}}$ is a prescribed physical nodal displacement vector, and $\boldsymbol{F}_{\mathrm{L}}$ and $\boldsymbol{F}_{\mathrm{NL}}$ are the linear and nonlinear contributions to the total nodal force.

$\boldsymbol{F}_{\mathrm{L}}$ is first obtained by prescribing $\boldsymbol{X}_{\mathrm{c}}$ in the linear static solution. $\boldsymbol{F}_{\mathrm{T}}$ is then obtained by prescribing $\boldsymbol{X}_c$ in the nonlinear static solution. Finally, the nonlinear contribution $\boldsymbol{F}_{\mathrm{NL}}$ is obtained by subtracting $\boldsymbol{F}_{\mathrm{L}}$ from $\boldsymbol{F}_{\mathrm{T}}$, or

$$\boldsymbol{F}_{\mathrm{NL}} = \boldsymbol{\Gamma}\left(\boldsymbol{X}_{\mathrm{c}}\right) = \boldsymbol{F}_{\mathrm{T}} - \boldsymbol{F}_{\mathrm{L}} \tag{2-11}$$

To illustrate the technique, one can begin by prescribing the displacement fields

$$\begin{aligned}
\boldsymbol{X}_{\mathrm{c}} &= +\boldsymbol{\phi}_1 q_1\\
\boldsymbol{X}_{\mathrm{c}} &= -\boldsymbol{\phi}_1 q_1
\end{aligned} \tag{2-12}$$

The nonlinear nodal force contributions $\boldsymbol{F}_{\mathrm{NL}}$ are determined using (7) after solving the linear and nonlinear static solutions. These may be written in modal coordinates as

$$\begin{aligned}\widetilde{\boldsymbol{F}}_{\mathrm{NL}_1} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{F}_{\mathrm{NL}_1} = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Gamma}\left(+\boldsymbol{\phi}_1 q_1\right) = \left[a_{11}^r\right] q_1 q_1 + \left[b_{111}^r\right] q_1 q_1 q_1 \\ \widetilde{\boldsymbol{F}}_{\mathrm{NL}_2} &= \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{F}_{\mathrm{NL}_2} = \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Gamma}\left(-\boldsymbol{\phi}_1 q_1\right) = \left[a_{11}^r\right] q_1 q_1 - \left[b_{111}^r\right] q_1 q_1 q_1\end{aligned} \tag{2-13}$$

where the sought stiffness coefficients $[a_{11}^r]$ and $[b_{111}^r]$ are vectors of length L. Note that the other nonlinear terms do not appear in (2-13) since $q_j = 0$ for $j \neq 1$. Since q_1 is a known scalar, the coefficients $[a_{11}^r]$ and $[b_{111}^r]$ for $r = 1, 2, \ldots, L$ can be determined from the resulting system (2-13) of 2 linear equations. The remaining coefficients $\left[a_{jj}^r\right]$ and $\left[b_{jjj}^r\right]$ $j = 1, 2, \ldots, L$ can be determined in an analogous manner." (Muravyov and Rizzi, 2003 [7])

In the following sections, the notation introduced by Muravyov and Rizzi is used.

2-4 In-plane modes

Structures with high aspect ratio like strings, beams, plates and membranes exhibit the special property of having a very low stiffness in (at least) one direction compared to the others. It turns out that the nonlinear ROM containing eigenmodes (corresponding to the linearised system) selected on frequency and load mapping criteria, is not capable to represent loadcases in which large displacements occur. To illustrate this, consider the following case:

For a simple fixed free beam, the fundamental eigenmode contains only significant out-of-plane (transversal) displacements. The in-plane (lateral) component is negligible. However, when a tip force is applied such that the resulting vertical displacement is in the order of the beam thickness, the tip also displaces horizontally. Therefore, the fundamental eigenmode on it's own cannot represent the true behaviour. When only including this mode in the ROM, it represents a structure which has constrained horizontal displacement at the free end. Therefore, the (nonlinear) ROM is stiffer than the real structure it should represent.

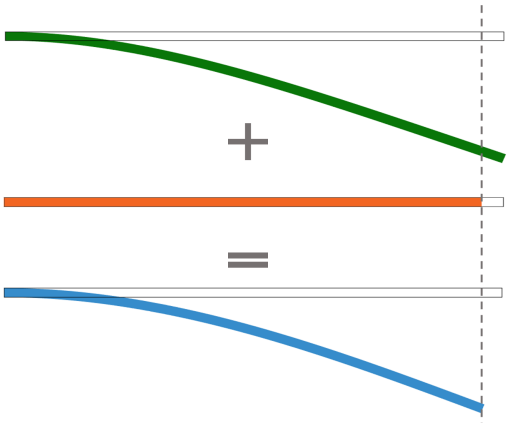


Figure 2-4: Summation of in and out-of-plane modes

ROM Completeness

From the previous it can be concluded that for large deformations and if e.g. in-plane dynamic eigenmodes or other modes are neglected, a ROM can be stiffer than the original structure. As a consequence, the linear eigenmodes (green in Figure 2-4) do not properly represent the large dynamic displacement field (blue in Figure 2-4).

If this is the case, softening shapes (or in-plane modes, like the orange shape in Figure 2-4), should be determined which could be added to make the ROM proper. The determination of those softening shapes turns out to be quite involved, and is discussed in the next section.

2-4-1 In-plane mode selection

Literature

In literature, some methods to determine softening in-plane vectors (like the orange shape in Figure 2-4) are reported. Here, two methods are mentioned.

The first method, described in [8], is based on the response to loadcases. A load in the shape of the out-of-plane eigenmode (green mode in Figure 2-4) is applied, and from the obtained displacement field the out-of-plane eigenvector (green mode in Figure 2-4) is subtracted. The remaining displacement vector (which should be the orange shape in Figure 2-4) is due to nonlinear coupling, and should contain the most important (having the largest softening effect) in-plane mode.

However, beside the desired in-plane mode, the resulting displacement field (orange shape in Figure 2-4) could also contain other in-plane or out-of-plane eigenvectors. This makes the procedure quite complicated, especially when the ROM contains multiple out-of-plane modes which all need additional in-plane modes.

A second approach [9] is based on the change of the eigenvectors along deformation in a certain mode. To determine this, derivatives are calculated with respect to the system matrices. In our scenario where the FE software is treated as a black box, the nonlinear stiffness matrix is not available. Therefore, this class of approaches cannot be used.

Besides, both above mentioned methods have an additional disadvantage. Both approaches yield additional shapes which are no eigenvectors of the structure, so they are not ideal to model in-plane dynamics. Therefore, in the following part, another method is introduced.

In-plane mode selection method

Above mentioned approaches construct softening shapes from scratch. However, the full set of eigenvectors still contains the complete linearised mechanics of the structure. Therefore, an eigenmode which describes in-plane displacement could be added to the ROM. This mode can decrease the effective stiffness of the out-of-plane eigenmode (having a softening effect). Beside, as it is strongly coupled, its dynamics can also play a role, possibly influencing effective inertia and damping. The idea of using still a linear basis of eigenvectors is already mentioned in [8], in the context of other approaches.

To automatically select the relevant in-plane mode(s), we make use of the softening effect of an in-plane mode on an out-of-plane mode. This can be determined directly from the governing equations (Eq. (2-3)), and requires only the very first stage of the STEP method (described in Eq. (2-13)).

Determining softening effect

In order to determine the softening effect of an in-plane mode on the out-of-plane (OOP) eigenmode, the first step is to apply a displacement field in the shape of the OOP eigenmode. The question for determining the importance of a mode is: If, additional to the applied shape, the applied field would contain a fraction of the in-plane mode, such that the in-plane modal force vanishes, would the OOP modal reaction force change? To answer this, consider the following sub-steps:

1. Prescribe a displacement field $\boldsymbol{u}$ in the shape of OOP mode $\boldsymbol{\phi}_1$ and evaluate the OOP modal force $f_1^a = \boldsymbol{\phi}_1^T \boldsymbol{F}(\boldsymbol{u_1})$ (red arrows in Figure 2-5-b) and in-plane modal force $f_2 = \boldsymbol{\phi}_2^T \boldsymbol{F}(\boldsymbol{u_1})$ (yellow arrows)
2. Add a fraction α of in-plane mode $\boldsymbol{\phi}_2$ to the prescribed displacement field $\boldsymbol{u}$, such that the in-plane modal force f_2 vanishes (Figure 2-5-d)
3. Compare the resulting OOP modal forces f_1^a and f_1^b (As can be seen, the vertical arrows in Figure 2-5-d are smaller than those in Figure 2-5-b.) From this, determine the softening effect of in-plane mode $\boldsymbol{\phi}_2$ on OOP mode $\boldsymbol{\phi}_1$.

The sub-steps described above can be performed numerically, using the information obtained from the first stage of the STEP method (Eq. (2-13)). This is explained in the next section.

Softening calculation

In this section, a mathematical description is provided to determine in-plane modes. The complete derivation can be found in section A-2. To explain the method, a very simple 2dof system is considered, although the method is also applicable to larger systems. Note that the derived formula itself was already proposed in literature [10], but not used in combination with the STEP method for the selection of in-plane modes.

Consider the static equations coupling two modes (from Eq. (2-3) and Eq. (2-9)):

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + a_{12}^{(1)} q_1 q_2 + a_{22}^{(1)} q_2^2 + b_{111}^{(1)} q_1^3 + b_{112}^{(1)} q_1^2 q_2 + b_{122}^{(1)} q_1 q_2^2 + b_{222}^{(1)} q_2^3 = F^{(1)} \tag{2-14}$$

$$k_2 q_2 + a_{11}^{(2)} q_1^2 + a_{12}^{(2)} q_1 q_2 + a_{22}^{(2)} q_2^2 + b_{111}^{(2)} q_1^3 + b_{112}^{(2)} q_1^2 q_2 + b_{122}^{(2)} q_1 q_2^2 + b_{222}^{(2)} q_2^3 = F^{(2)} \tag{2-15}$$

$$\tag{2-16}$$

When applying the complete STEP method for two modes, all coupling terms could be determined. However, in this way, it would require a huge computational cost to investigate the softening effect of all modes on one OOP mode. For a system consisting of 100 eigenmodes, the full STEP method should be processed 99 times to investigate the softening effect of all modes independently on one OOP mode.

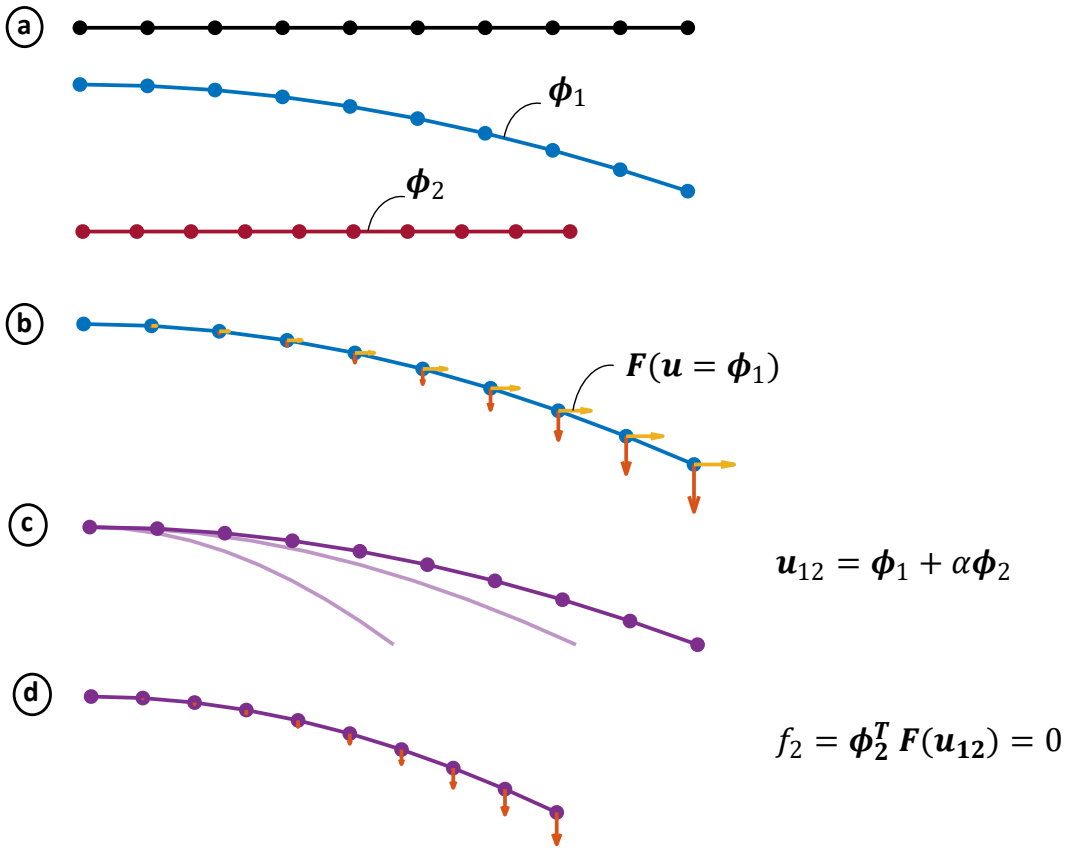


Figure 2-5: In-plane mode selection method for a fixed-free beam: a) undeformed configuration, out-of-plane eigenmode ϕ_1 and in-plane eigenmode ϕ_2 b) reaction forces due to subjected displacement field $u = \phi_1$ c) combine in-plane and out-of-plane eigenmodes to construct a displacement field d) combination of eigenmodes such that the in-plane modal force equals zero

To reduce computational cost, an approximation is constructed. This approximation requires only one iteration of the very first stage of the STEP method (Eq. (2-13)) to determine the softening effect of all modes on a certain OOP mode. Consider the following approach:

Processing the very first stage of the STEP method yields the $a_{11}^{(i)}$ and $b_{111}^{(i)}$ coefficients. Using the relations among coefficients (like $a_{12}^{(1)} = 2a_{11}^{(2)}$, derived in section C-2-1, Eq. (C-15)) and removing all terms with unknown coefficients yields:

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + a_{12}^{(1)} q_1 q_2 + b_{111}^{(1)} q_1^3 + b_{112}^{(1)} q_1^2 q_2 = F^{(1)} \tag{2-17}$$

$$k_2 q_2 + a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3 = F^{(2)} \tag{2-18}$$

For a given q_1, an estimate for q_2 can be calculated (by solving for $F^{(2)} = 0$ like in Figure 2-5-d):

$$q_2 = -\frac{a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3}{k_2} \tag{2-19}$$

Eq. (2-19) can be substituted into Eq. (2-17) yielding (the complete derivation can be found in section A-2):

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + b_{111}^{(1)} (1 - \vartheta_2^{(1)}) q_1^3 = F^{(1)} \tag{2-20}$$

$$\vartheta_2^{(1)} = 2 \frac{\left(a_{11}^{(2)}\right)^2}{b_{111}^{(1)} k_2} \tag{2-21}$$

Eq. (2-20) can be seen as an approximation of Eq. (2-14) by eliminating Eq. (2-16). Here, $\vartheta_i^{(j)}$ can be seen as the relative influence of the in-plane mode i on the out-of-plane mode j, and thus acts as a criteria to determine whether the in-plane mode should be incorporated into the ROM or not. Note that for all modes, the a and b coefficients needed to calculate $\vartheta_i^{(j)}$ for one OOP mode are available from only applying two displacement fields to the structure (the very first stage of the STEP method).

As mentioned before, the idea of including linear in-plane eigenvectors is not entirely new [8], and the derived formula (Eq. (2-20)) was already reported in [10] (although not in the context of in-plane mode selection). However, in this work, it is implemented for the STEP method in a computationally efficient way, which is new as far as known to the author.

Limitations

The previous presented method was implemented and tested for strings and circular membranes (results are shown in section 3-1-3). The method works fine as long as only one or a small number of in-plane modes are needed per out-of-plane mode. For strings and membranes, this is the case. However, for other structures, this does not necessary have to be the case, possibly yielding a very large number of modes in the ROM. This is undesirable, as it increases calculation time of both the STEP method and the post-processing. Therefore, in the next section, another way to deal with in-plane modes is presented.

2-4-2 Static condensation

In the previous section, a method was presented to explicitly include in-plane modes in the ROM. In this way, inertia, damping and stiffness of in-plane modes are fully included in the nonlinear ROM. However, if the dynamics of in-plane modes are not of particular interest, for some structures, they can be neglected.

Conditions

Neglecting in-plane dynamics is allowed as long as the in-plane modes do not interact with relevant out-of-plane modes, which can be assumed when the in-plane eigenfrequencies are far above the out-of-plane eigenfrequencies. In this case, in-plane modes operate far below their eigenfrequency and thus their effective mass can be neglected. As their stiffness comprises a

softening effect on the out-of-plane modes, this cannot be neglected as will be explained later on.

The simplification can be made for structures where all in-plane displacements can be described by a fixed set of dofs, while the remaining dofs are still able to represent all eigenmodes in the ROM.

Approach

To apply static condensation, it should be determined for each dof of the complete system whether it represents out-of-plane or in-plane displacement. The dof corresponding to the in-plane direction are remained free in the STEP procedure, such that the solver can search for the in-plane displacement which results in the lowest in-plane modal forces (as shown in Figure 2-5 and Eq. (2-19)). In this way, in-plane stiffness (resulting in softening) is still taken into account, while in-plane mass is neglected.

As the in-plane modes are free to respond (condense) instantaneously (as their mass is neglected), they act like static springs, and the method is called static condensation. For example strings, membranes, flat plates and straight beams are well suited for this simplification method.

An example

As an example, consider the structure from Figure 2-3 again: A 2D cantilever consisting of 2 nodes (shown in Figure 2-6), each having only translational degrees of freedom.

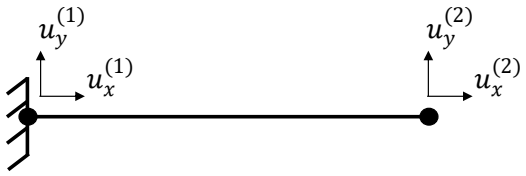


Figure 2-6: A simple cantilever consisting of 2 nodes.

The displacement vector $\mathbf{u}$ writes:

$$\mathbf{u}^{\mathrm{T}} = [u_{\mathrm{x}}^{(1)} u_{\mathrm{y}}^{(1)} u_{\mathrm{x}}^{(2)} u_{\mathrm{y}}^{(2)}] \tag{2-22}$$

Here, the superscript (1) indicates the left fixed node, and (2) the free end. Now, an important step is to identify the directions of in and out-of-plane modes. For the simple cantilever shown in Figure 2-3, in-plane modes are assumed to act in horizontal direction (x), while OOP modes are assumed to act in vertical (y) direction. Consider the following eigenvector:

$$\boldsymbol{\phi}_1^{\mathrm{T}} = [0\,0\,0\,1] \tag{2-23}$$

which corresponds to vertical displacement of the free tip. If this would be used in the STEP method, the tip would be constrained in x-direction because $u_{\mathrm{x}}^{(2)} = 0$ in $\boldsymbol{\phi}_1$. This corresponds

to the green deformation in Figure 2-4. However, in line with the static condensed STEP method, only dof corresponding to the OOP direction are prescribed. Therefore, the vector containing prescribed displacements should contain only the following dof:

$$\mathbf{x}^{\mathrm{T}} = [u_{\mathrm{y}}^{(1)} u_{\mathrm{y}}^{(2)}] \tag{2-24}$$

When using eigenvector $\boldsymbol{\phi}_1$ in the STEP process, the corresponding prescribed displacement field writes:

$$\mathbf{x}^{\mathrm{T}} = [u_{\mathrm{y}}^{(1)} u_{\mathrm{y}}^{(2)}] = [0\ 1] \tag{2-25}$$

Now, it is clear that only the OOP component of the eigenvector is prescribed, while the in-plane degrees of freedom are left free. In this way the solver can determine the horizontal tip displacement corresponding to the lowest in-plane force (as it searches for a minimum in potential energy).

Advantages

In the code written, this method is also implemented. Appendix section F-1-4 shows part of the code associated to this method (note that in line 12 the dof corresponding to the OOP direction are selected). The computational cost of the STEP method is significantly reduced, as at least half the number of modes are needed in the ROM (compared when including in-plane modes explicitly), and no in-plane mode selection procedure is needed. Furthermore, the resulting set of equations is much smaller, reducing computation time for post-processing (like bifurcation analysis).

Literature

In literature, the same concept of static condensation is applied for an analytic approach analysing a string [1]. Besides, the term static condensation is mentioned in the context of post-processing [8] (ODE solvers etc.). As far as known to the author, static condensation is not applied yet for the STEP method.

If the structure does not fulfil the conditions to use static condensation, another variant of the STEP method, discussed in the next section, could possibly be used.

2-4-3 Compliant ROM

For the mentioned approaches thus far, all modes that are not included in the ROM were fixed. However, when in reality the structure is subjected to a force, it is free to move in those modes. The STEP procedure can be changed such that it is compliant in all excluded modes if a force instead of displacement field is applied. By including constraints, resulting modal displacements of included modes can be managed.

This method was implemented for a string, to investigate the feasibility. Compared to the condensed STEP variant, it was slightly slower but didn't need any information of in-plane directions.

Literature

The same concept (although different applied) is mentioned in literature [11] and [12], in the context of the *'applied loads method'*. However, this method does not use constraints to manage mode contributions in the response. Instead, regression is used to determine nonlinear stiffness from several loadcases, which should in general be less accurate.

2-5 Implementation

To implement the STEP method, a FEM package as well as a programming environment for pre and post processing is needed. The FEM package should have the following features:

- modal analysis
- geometric nonlinear static analysis
- interfacing
- parallelization (preferred)

Comsol is chosen as the best candidate, to be combined with MATLAB, as those have a well developed interface. MATLAB is well suited for pre- and post processing. For bifurcation analysis of large systems, AUTO is more suited, which can interface with MATLAB by text files.

2-5-1 Overview

The programming structure is summarized in figure 2-7.

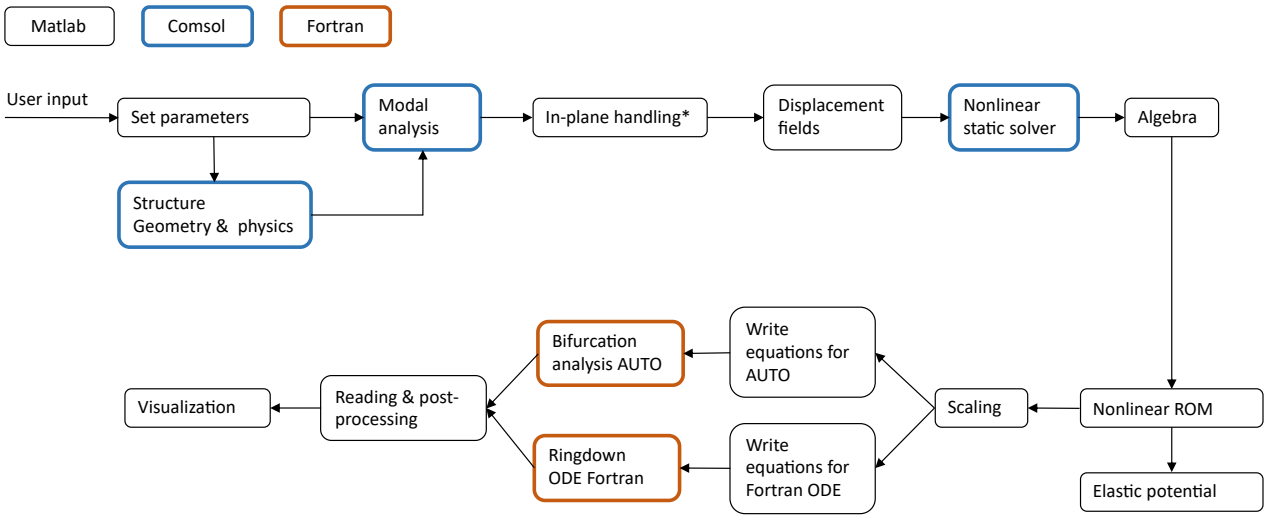


Figure 2-7: Overview code structure. * In-plane handling: choose between explicit including in-plane-modes (section 2-4-1) and static condensation (section 2-4-2).

2-5-2 Matlab-Comsol interface

The Matlab-Comsol interface is build from (among others) the following parts:

- in Comsol:
  - A component, containing parametrized geometry, physics and mesh
  - A study containing an eigenvalue solver
  - A study containing a static nonlinear solver
- in Matlab:
  - A script declaring all structure parameters and solver settings (appendix F-1-1)
  - A script interfacing for modal analysis (appendix F-1-5)
  - A script creating displacement fields (appendix F-1-3)
  - A script interfacing with the nonlinear solver (appendix F-1-6 and F-1-2)
  - A script performing linear algebra to determine the nonlinear stiffness (appendix F-1-7)

The implementation of the STEP procedure can be summarized by the following sequence:

1. In Matlab, structure parameters and solver settings are provided as input
2. Through the Application programming interface (API), this is send to Comsol
3. The eigenvalue analysis in Comsol is started (from API)
4. All structure information is gathered from Comsol (Mesh, dofs, system matrices)
5. All solution information (eigenvectors & eigenvalues) is gathered.
6. The framework needed for the prescribed displacements is built from the ROM settings. Optionally, static condensation is applied ór additional in-plane modes are selected.
7. The (scaled) eigenvectors are loaded in Comsol by interpolation functions (which are later on used to construct prescribed displacement fields)
8. A Parametric sweep is set using the API, which defines for each loadcase the contribution of each eigenvector (using the framework)
9. All solutions (nodal reaction forces) are loaded in Matlab, to determine the modal forces
10. Using linear algebra, all stiffness coefficients are determined (using Eq. (2-13) and similar equations described in [7])

Elements

In Comsol, different structural elements are available. Most parts of the code are independent of the element type used. However, as different elements generate different output, a small piece of code interfacing with Comsol is element specific. In Table 2-1 the elements for which the interface code is build and tested are listed. For every element, their linear stiffness is listed, beside the nonlinear stiffness the code can extract. In the chapter 3, the nonlinear stiffness is tabulated and compared to values obtained in literature from analytical methods (as mentioned in section 2-2-1).

Using truss and membrane elements, nonlinear stiffness due to stretching of the mid-plane can be identified. This can be a quite easy detectable in practice. However, using beam and plate

Table 2-1: Stiffness contributions for different element types in Comsol. All listed elements were implemented and tested, but other elements might also be used. Even for different types of multiphysics the presented methodology might be employed (optical or electromagnetic resonators). Note that the beam element in Comsol is not suited for our type of nonlinear analysis.

	Linear stiffness			Nonlinear stiffness	
Element	Axial	Pre-stress	Bending	Mid-plane stretching	Bending
2D truss	✓	✓	×	✓	×
2D beam	✓	×	✓	×	×
3D membrane	✓	✓	×	✓	×
2D plate	✓	✓	✓	✓	✓

Table 2-2: Implemented methods for different element types. Note that all analysed structures are flat (2D).

Element	Condensed STEP	In-plane STEP	Compliant STEP
2D truss	✓	✓	✓
3D membrane	✓	✓	
2D plate	✓	✓	

elements, also nonlinear bending stiffness can be identified. For example, the plate element can model a fully-clamped pre-stressed plate, having linear bending and pre-stress stiffness, and nonlinear stiffness consisting of both mid-plane stretching and nonlinear bending.

2-5-3 Matlab-AUTO interface

In order to study forced vibrations, 'AUTO 97: Continuation and bifurcation software for ordinary differential equations' by E.J. Doedel is used. This software is able to calculate frequency response curves for nonlinear systems, and detects most bifurcations points. It can also be used for parametric excitation, but it is not suited for random excitation or discontinuous functions. As it is written in Fortran, it is very fast, and large systems are allowed (say up to 20 2nd order Ordinary differential equation (ODE)'s). The interface can be summarized as follows:

1. The system of 1^{st} order ODEs is written by Matlab into a text file.
2. The solver settings are written by Matlab in separate text files.
3. In Matlab, using the API of the compiler, the AUTO project workspace (including libraries, settings and ODE files) is compiled, generating an executable.
4. In Matlab, the executable AUTO file is run, performing the bifurcation analysis
5. The output filed generated by the executable (containing the solutions) are red by Matlab

This scheme was partly implemented by Koen Markestein before, although it was coded in Python and step 3 (automated compilation) was not included yet. Apart from that, automated compilation was also used by Justin Smid, although in a more basic way. Some notes on the use of AUTO can be found in appendix A-1-2.

2-5-4 Matlab-Fortran ODE

In order to study free vibrations (including noise), time integration is used. For large, strongly nonlinear systems, the ODE solvers in Matlab are way too slow. Therefore, again Fortran is used to perform the time integration. Fortran contains International Mathematics and Statistics Library (IMSL) libraries containing ODE solvers well suited for large nonlinear systems. The subroutine DIVPAG is called, which is especially suited for stiff equations as it is set to use Gear's Backward differentiation formula (BDF) method. The interface can be summarised as follows:

1. The system of 1st order ODE's, the settings and governing Fortran code is written by Matlab into a text file
2. Initial conditions are written by Matlab in a text file
3. The project workspace is compiled, generating an executable
4. By Matlab, the executable file is run, performing time integration
5. The output filed generated by the executable (containing the solutions) are loaded into Matlab

For accuracy, the total energy in the system is monitored, yielding admissible tolerance settings. The created executables can be run stand-alone (only need the initial conditions file). Therefore, they can be run in parallel, using the full capacity of all CPU's and making it ideal for cluster computing. The Fortran script was supplied by F. Aljiani, and was changed slightly to make it suitable for our purpose.

2-5-5 Scaling

In order to use the ROM in ODE solvers or bifurcation analysis software, the equations should be scaled in a proper way. Most software packages cannot handle extraordinary large or small numbers, and perform best when the equation contains terms in the range of $\mathcal{O}(1)$. To scale the ROM, the eigenvectors, time and space can be scaled. In the following, it is shown how scaling is implemented in this work.

Time and space scaling

The eigenvectors are scaled such that all equations of motion have unit mass. Then, time and space are scaled such that for the first eigenmode in the ROM, unit linear and nonlinear stiffness are obtained.

Consider the mass-normalised equation of motion of the first mode:

$$\ddot{u} + ku + au^2 + bu^3 + 2\zeta\omega_n\dot{u} = f\sin(\omega t) \tag{2-26}$$

Space and time are scaled the following way (where h and T are defined in Eq. (2-29)):

$$\tilde{u} = \frac{u}{h}, \qquad \tau = \frac{t}{T} \tag{2-27}$$

yielding (see appendix A-2-1)

$$\tilde{u}'' + \tilde{k}\tilde{u} + \tilde{a}\tilde{u}^2 + \tilde{b}\tilde{u}^3 + 2\zeta\tilde{\omega}_n\tilde{u}' = \tilde{f}\sin(\tilde{\omega}\tau) \tag{2-28}$$

For good numerical performance, we impose $\tilde{k} = 1$ and $\max\{\tilde{a}, \tilde{b}\} = 1$. Therefore,

$$T = \frac{1}{\sqrt{k}}, \qquad h = \min\left\{\frac{k}{a}, \sqrt{\frac{k}{b}}\right\} \tag{2-29}$$

Eigenvector scaling - nonlinear stiffness

For linear systems, eigenvectors can have arbitrary scaling, and modal stiffness and mass can be calculated easily. If later on another scaling is preferred, modal quantities can be recalculated. For systems having nonlinear terms however, rescaling the eigenvectors does not result in straightforward rescaling of the nonlinear terms. The following part describes a procedure for rescaling the reduction basis. The full derivation can be found in appendix A-2-2.

Mass-normalised quantities are denoted with subscript m, quantities corresponding to max-1 displacement eigenvectors are denoted with subscript u. The modal transformation for one eigenmode is defined as $\mathbf{x} = \boldsymbol{\phi} q$. The ratio between the eigenvectors is defined as:

$$\boldsymbol{\phi}_{\mathrm{m}} = \boldsymbol{\phi}_{\mathrm{u}}\alpha \tag{2-30}$$

Yielding for each eigenmode i:

$$q_{\mathrm{m}_i} = \frac{q_{\mathrm{u}_i}}{\alpha_i} \tag{2-31}$$

Consider γ as the nonlinear modal force term, depending on the modal displacements included in the ROM: $\gamma(q_1, q_2, ..., q_L)$.

For one mode i, the equation of motion writes:

$$\boldsymbol{\phi}_{\mathrm{m}_i}^T\mathbf{M}\boldsymbol{\phi}_{\mathrm{m}_i}\ddot{q}_{\mathrm{m}_i} + \boldsymbol{\phi}_{\mathrm{m}_i}^T\mathbf{K}\boldsymbol{\phi}_{\mathrm{m}_i}q_{\mathrm{m}_i} + \gamma_{\mathrm{m}_i}(q_{\mathrm{m}_1}, q_{\mathrm{m}_2}, ...) = \boldsymbol{\phi}_{\mathrm{m}_i}^T\mathbf{F} \tag{2-32}$$

This can be rewritten in terms of max-1 based eigenmodes (appendix A-2-2):

$$\boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{u}_i} \ddot{q}_{\mathrm{u}_i} + \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{K} \boldsymbol{\phi}_{\mathrm{u}_i} q_{\mathrm{u}_i} + \gamma_{\mathrm{u}_i}(q_{\mathrm{u}_1}, q_{\mathrm{u}_2}, ...) = \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{F} \tag{2-33}$$

Here, γ is rewritten such that the correction terms are implemented in the coefficients of all nonlinear terms. For each nonlinear stiffness term, we obtain:

$$b_{\mathrm{u}_{ijk}}^{(r)} = \frac{b_{\mathrm{m}_{ijk}}^{(r)}}{\alpha_r \alpha_i \alpha_j \alpha_k}, \qquad a_{\mathrm{u}_{ij}}^{(r)} = \frac{a_{\mathrm{m}_{ij}}^{(r)}}{\alpha_r \alpha_i \alpha_j} \tag{2-34}$$

Here, (r) represents the equation index, such that $b_{\mathrm{u}_{ijk}}^{(2)}$ corresponds to the second equation of motion (in line with the notation introduced in section 2-3-3). Max-1 based modal linear quantities can be directly obtained from multiplication with the max-1 based eigenvectors.

Chapter 3

Dynamics

In this chapter, several simple structures are analysed using the methods explained in the previous chapter. First, resulting nonlinear stiffnesses are non-dimensionalised and tabulated. Next, each structure is analysed by a frequency sweep, revealing nonlinear dynamic effects. Successively, two structures of which experimental data is available are simulated, followed by the free vibrations to investigate FPU behaviour.

3-1 Dynamics of simple structures

3-1-1 Equations of motion

Recall the set of coupled modal equations (Eq. (2-3)), obtained after applying the modal transformation:

$$\widetilde{\boldsymbol{M}}\ddot{\boldsymbol{q}} + \widetilde{\boldsymbol{C}}\dot{\boldsymbol{q}} + \widetilde{\boldsymbol{K}}\boldsymbol{q} + \gamma\left(q_1, q_2, \ldots, q_L\right) = \widetilde{\boldsymbol{F}}$$

where the nonlinear force vector could be expanded in the following form (2-9):

$$\gamma_r\left(q_1, q_2, \ldots, q_L\right) = \sum_{j=1}^{L}\sum_{k=j}^{L} a_{jk}^{r} q_j q_k + \sum_{j=1}^{L}\sum_{k=j}^{L}\sum_{l=k}^{L} b_{jkl}^{r} q_j q_k q_l, \qquad r = 1, 2, \ldots, L$$

For a ROM consisting of two modes, the (undamped) equations of motion write:

$$\begin{aligned} m_1\ddot{q}_1 + k_1 q_1 + a_{11}^{(1)} q_1^2 + a_{12}^{(1)} q_1 q_2 + a_{22}^{(1)} q_2^2 + b_{111}^{(1)} q_1^3 + b_{112}^{(1)} q_1^2 q_2 + b_{122}^{(1)} q_1 q_2^2 + b_{222}^{(1)} q_2^3 &= F^{(1)} \\ m_2\ddot{q}_2 + k_2 q_2 + a_{11}^{(2)} q_1^2 + a_{12}^{(2)} q_1 q_2 + a_{22}^{(2)} q_2^2 + b_{111}^{(2)} q_1^3 + b_{112}^{(2)} q_1^2 q_2 + b_{122}^{(2)} q_1 q_2^2 + b_{222}^{(2)} q_2^3 &= F^{(2)} \end{aligned} \tag{3-1}$$

which just fits in one line. However, to quickly compare values, it is more convenient to put them into tables, like Table 3-1. Here, each column contains the coefficients present in one

equation of motion. The shown values are non-dimensionalised with factors a_{dim}, so they can be used for arbitrary parameter values. The value of each coefficient can be determined as:

$$a_{11}^{(1)} = \tilde{a}_{11}^{(1)} a_{\text{dim}} \tag{3-2}$$

and the same notation holds for the cubic coefficients.

Table 3-1: Pre-factors for quadratic coefficients, such that $a_{11}^{(1)} = \tilde{a}_{11}^{(1)} a_{\text{dim}}$

	Eq. 1	Eq. 2
a_{11}	$\tilde{a}_{11}^{(1)}$	$\tilde{a}_{11}^{(2)}$
a_{12}	$\tilde{a}_{12}^{(1)}$	$\tilde{a}_{12}^{(2)}$
a_{22}	$\tilde{a}_{22}^{(1)}$	$\tilde{a}_{22}^{(2)}$

3-1-2 String

For a simple, straight string, the STEP method is applied using static condensation (so in-plane modes are not explicitly present) for a ROM including the three first eigenmodes. Modal linear coefficients are listed in Table 3-2, and the non-dimensionalising terms for both linear and nonlinear quantities are listed in Table 3-3. Non-dimensionalised quadratic stiffness coefficients are shown in Table 3-4 (here, they all equal zero), the cubic terms are shown in Table 3-5.

Table 3-2: Non-dimensionalised linear modal values

	Eq. 1	Eq. 2	Eq. 3
m/m_{t}	0.5	0.5	0.499
k/k_0	4.93	19.7	44.4
$\omega_{\text{n}}/\omega_0$	1	2	3

Table 3-3: Scaling values string

Linear			Nonlinear	
m_{t}	k_0	ω_0	a_{dim}	b_{dim}
ρAL	$\frac{\sigma A}{L}$	$\frac{1}{2L}\sqrt{\frac{\sigma}{\rho}}$	$\frac{EA}{L^2}\frac{\delta h}{L}$	$\frac{EA}{L^3}$

In literature [1], the same value for $b_{111}^{(1)}$ (both $\tilde{b}_{111}^{(1)}$ and b_{dim}) is reported. The other coupling therms are not calculated in [1], but [2] proposes a method to determine them using a perturbation technique (this source is a bit less comprehensible).

From the tables (Table 3-2, Table 3-3, Table 3-4 & Table 3-5), the following can be said about straight strings:

- All eigenmodes have equal modal mass

Table 3-4: Non-dimensionalised quadratic coefficients

	Eq. 1	Eq. 2	Eq. 3
a_{11}	0	0	0
a_{12}	0	0	0
a_{13}	0	0	0
a_{22}	0	0	0
a_{23}	0	0	0
a_{33}	0	0	0

Table 3-5: Non-dimensionalised cubic coefficients straight string

	Eq. 1	Eq. 2	Eq. 3
b_{111}	12.2	0	0.107
b_{112}	0	49.4	0
b_{113}	0.417	0	111
b_{122}	49.4	0	1.62
b_{123}	−0.0711	3.36	−0.132
b_{133}	111	0	0.0151
b_{222}	0	195	0
b_{223}	1.69	0	445
b_{233}	0	445	0
b_{333}	−0.002 98	0	989

- Eigenfrequencies are linear increasing and just equal the mode index
- All quadratic stiffness terms equal zero, no softening is present for a straight string.
- Even (2) and uneven (1 & 3) modes are hardly coupled by cubic stiffness. Even modes have the same amount of displacement in positive and negative vertical direction. For uneven modes, this is not the case, which could be the reason for the particular small coupling between even and uneven modes. Note that this doesn't hold for all terms, as for example $\tilde{b}^{(1)}_{122}$ is nonzero.
- As linear stiffness, nonlinear stiffness increases for increasing mode index. Because eigenmodes only differ in the number of waves, nonlinear stiffness terms seem to increase in a constant portion (e.g. $\tilde{b}^{(3)}_{223} = 4\tilde{b}^{(3)}_{113}$, although this relation is not set by the elastic potential (Eq. (C-15)))

The frequency response is shown in Figure 3-1 and Figure 3-2. All modes show hardening, and mode 3 (which is, like mode 1, asymmetric) shows small interaction with mode 1. However, this is negligible, as it remains invisible at the point of (numerical) measurement.

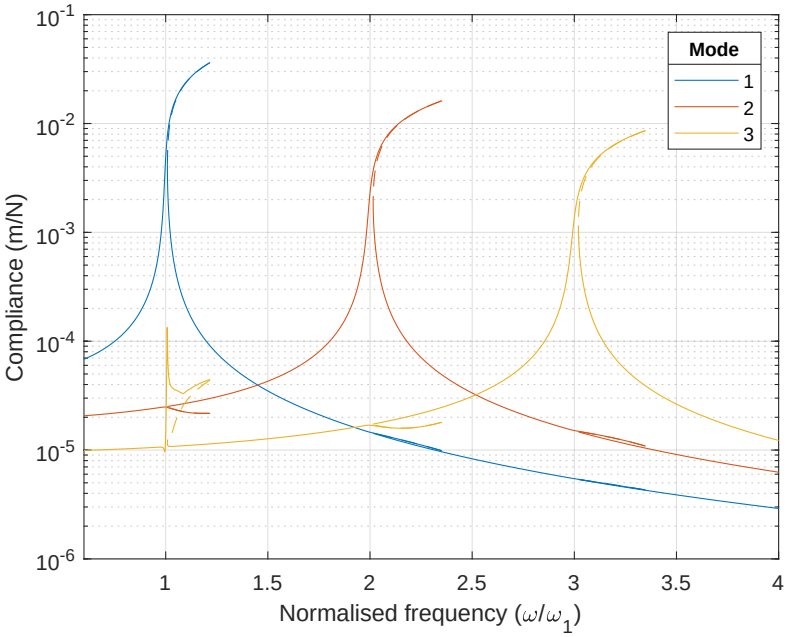


Figure 3-1: Modal frequency response string for external force of $0.6\,\mathrm{N}$ at $x_p = \frac{2}{3}L$. Dashed lines represent unstable solutions.

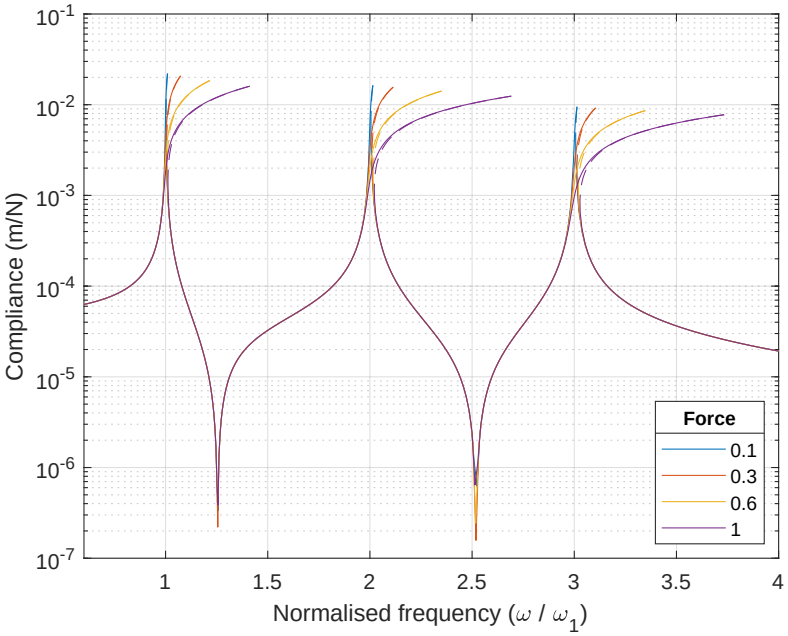


Figure 3-2: Frequency response of a string for increasing external force level. Excitation and measurement at the same point (collocated), $x_p = \frac{2}{3}L$. Dashed lines represent unstable solutions.

Peak value

In Figure 3-2, it is visible that the maximum compliance decreases with forcing amplitude (or absolute amplitude). However, the damping present in the equations remains constant, so the effect is purely due to the nonlinear hardening. To understand what's happening, the single dof duffing equation is analysed in the following part.

The forced Duffing equation including viscous damping writes:

$$\ddot{x} + \delta\dot{x} + \alpha x + \beta x^3 = \gamma\cos(\omega t) \tag{3-3}$$

Using the method of harmonic balancing, solutions can be found (shown in section B-1). The maximum compliance at resonance can be approximated as:

$$\frac{z_{\max}}{\gamma} \approx \frac{1}{\delta\omega} \tag{3-4}$$

which reveals that indeed, for increasing forcing (and thus eigenfrequency), the compliance decreases at the limit point. In section B-1 more details can be found.

3-1-3 String including in-plane modes

To show the resemblance between both variants of the STEP method (the first one using in-plane modes (section 2-4-1), the second using static condensation (section 2-4-2)), the analysis of a straight string is performed including in-plane modes in the ROM. The method described in section 2-4-1 is used to select the proper in-plane mode (shown in Figure 3-4) which has a softening effect on the out-of-plane mode. Only 1 out-of-plane eigenmode is included, just to show the principle. The resemblance between both methods is further investigated and analytically proven in appendix D-1. The possibility of interaction between the in-plane and out-of-plane modes of a string was also mentioned (but not elaborated) by [2].

Modal linear coefficients are listed in Table 3-7, the non-dimensionalising terms for both linear and nonlinear quantities are listed in Table 3-6. Non-dimensionalised quadratic stiffness coefficients are shown in Table 3-8, the cubic terms are shown in Table 3-9.

Table 3-6: Scaling values straight string of in-plane mode

Linear					Nonlinear	
m_{t}	$k_{0_{\mathrm{out}}}$	$k_{0_{\mathrm{in}}}$	$\omega_{0_{\mathrm{out}}}$	$\omega_{0_{\mathrm{in}}}$	a_{dim}	b_{dim}
ρAL	$\frac{\sigma A}{L}$	$\frac{EA}{L}$	$\frac{1}{2L}\sqrt{\frac{\sigma}{\rho}}$	$\frac{1}{L}\sqrt{\frac{E}{\rho}}$	$\frac{EA}{L^2}$	$\frac{EA}{L^3}$

Table 3-7: Non-dimensionalised linear modal values of a straight string including in-plane modes. Note the different scaling per equation: In Eq. 1, $k_0 = k_{0_{\text{out}}}$, and for Eq. 2, $k_0 = k_{0_{\text{in}}}$. This also holds for ω_0.

	Eq. 1	Eq. 2
m/m_t	0.5	0.5
k/k_0	4.93	4.97
ω/ω_0	1	2

Table 3-8: Non-dimensionalised quadratic coefficients of a straight string including in-plane modes

	Eq. 1	Eq. 2
a_{11}	0	7.75
a_{12}	15.5	0
a_{22}	0	0

Table 3-9: Non-dimensionalised cubic coefficients of a straight string including one in-plane mode.

	Eq. 1	Eq. 2
b_{111}	18.3	0
b_{112}	0	48.7
b_{122}	48.7	0
b_{222}	0	292

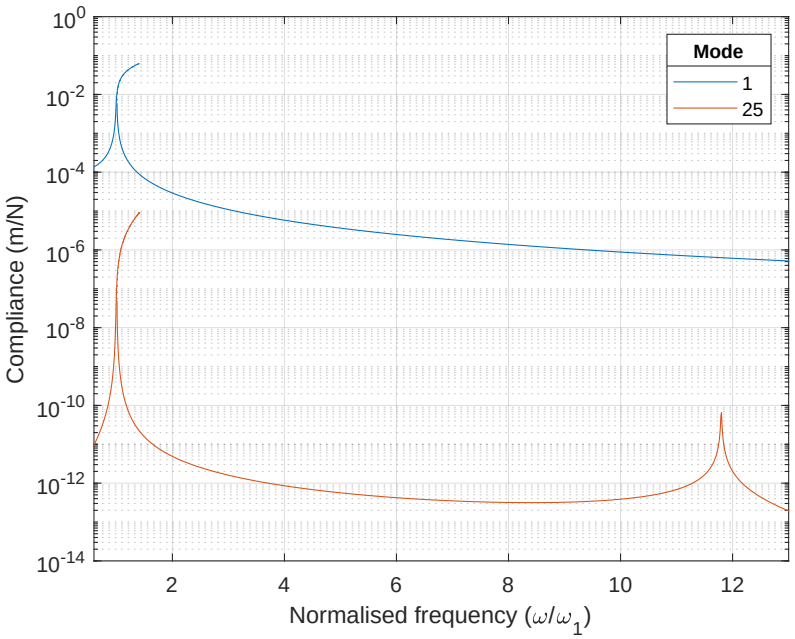


Figure 3-3: Modal frequency response of a string (including an in-plane mode) for external force of $1\,\mathrm{N}$ at $x_p = \frac{2}{3}L$.

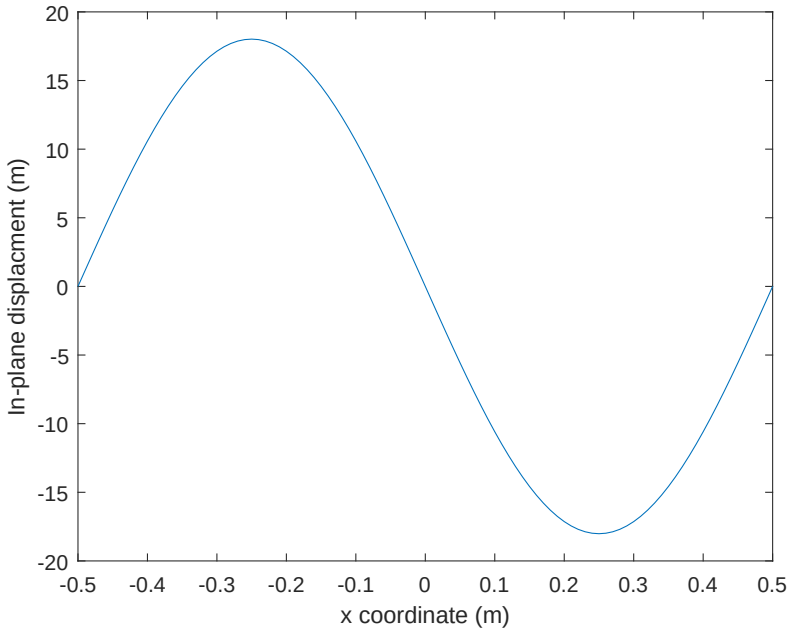


Figure 3-4: In-plane eigenmode coupled to the fundamental out-of-plane eigenmode.

Conclusion

Comparing $\tilde{b}^{(1)}_{111}$ values, it can be concluded that indeed the stiffness yielding from static condensation (Table 3-5) is lower than the one obtained when including in-plane modes (Table 3-9). This is correct according to Eq. (2-20), which predicts a lower $\tilde{b}^{(1)}_{111}$ stiffness if the in-plane mode is free to move (as it is during static condensation STEP). The effective stiffness of mode 1 in the ROM including in-plane modes is thus lower than the one reported in Table 3-9, which is revealed when loading the structure. This effect is further explained in appendix D-1. Further note that indeed the quadratic stiffness terms $\tilde{a}^{(2)}_{11}$ and $\tilde{a}^{(2)}_{11}$ are non-zero, allowing for the selection criterion stated in Eq. (2-21).

3-1-4 Circular membrane

As a second step, a slightly more complicated structure, a circular membrane (without bending stiffness), is analysed. The physical quantities describing its geometry and physics are listed in Table 3-10. Again, the STEP method is applied using static condensation (so in-plane modes are not explicitly present) for a ROM including the three non-degenerate eigenmodes. Modal linear coefficients are listed in Table 3-11, and the non-dimensionalising terms for both linear and nonlinear quantities are listed in Table 3-12. Non-dimensionalised cubic stiffness coefficients are shown in Table 3-13 (the quadratic terms appear to all equal zero).

Table 3-10: Properties circular membrane

Parameter	Formula	Units	Description
E	-	Pa	Young's modulus
h	-	m	Thickness
R	-	m	Radius
ν	-	-	Poisson's ratio
ρ	-	$\mathrm{kg/m^3}$	Density
σ	-	Pa	Stress
$\tilde{\rho}$	ρh	$\mathrm{kg/m^2}$	Mass per unit area
T_0	σh	$\mathrm{N\,m^{-1}}$	Membrane tension

Table 3-11: Non-dimensionalised linear modal values circular membrane. Index included modes: 1,2,6

	Eq. 1	Eq. 2	Eq. 3
m/m_t	0.269	0.244	0.115
k/k_0	4.9	11.2	11
$\omega_\mathrm{n}/\omega_0$	1	1.59	2.3

The frequency response of the circular membrane is shown in Figure 3-6 and Figure 3-7. Here, the first 9 non-degenerate modes are included in the ROM. The shape and eigenfrequency of those modes can be found in Table 3-14, simulation settings are listed in Table 3-15.

Table 3-12: Scaling values of a flat circular membrane [13]

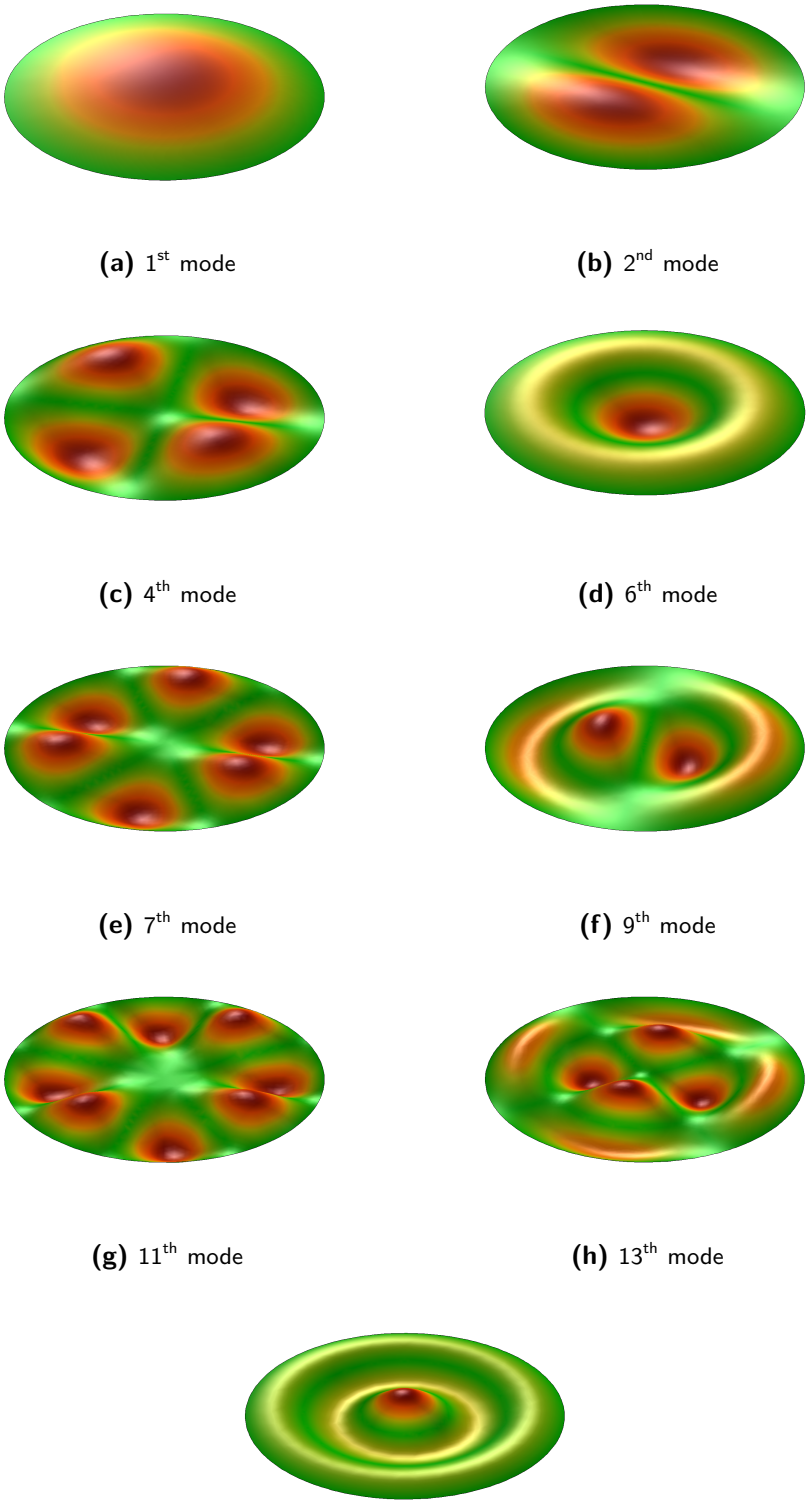
Linear			Nonlinear	
m_t	k_0	ω_0	a_{dim}	b_{dim}
$\pi R^2 h\rho$	T_0	$\frac{2.405}{R}\sqrt{\frac{T_0}{\tilde{\rho}}}$	$[-]$	$\frac{\pi}{1.27-0.97\nu-0.27\nu^2}\frac{Eh}{R^2}$

Table 3-13: Non-dimensionalised cubic coefficients circular membrane. Index included modes: 1,2,6

	Eq. 1	Eq. 2	Eq. 3
b_{111}	1	0.000 238	−0.201
b_{112}	0	3.32	0
b_{113}	−0.601	0	3.26
b_{122}	3.32	0.000 741	−0.252
b_{123}	0	−0.503	0.003 39
b_{133}	3.26	0.001 66	−3.45
b_{222}	0.000 19	7.19	−0.000 298
b_{223}	−0.252	−0.000 235	6.02
b_{233}	0.001 62	6.02	−0.003 01
b_{333}	−1.15	−0.001 28	8.06

Table 3-14: Normalised eigenfrequencies of a circular membrane

Index	Waveindex(t,r)	ω_i/ω_0	Shape
1	(0,1)	1	axisymmetric
2	(1,1)	1.5943	
3	(1,1)	1.5945	
4	(2,1)	2.1404	
5	(2,1)	2.1406	
6	(0,2)	2.3044	axisymmetric
7	(3,1)	2.6664	
8	(3,1)	2.6677	
9	(1,2)	2.9423	
10	(1,2)	2.9448	
11	(4,1)	3.1847	
12	(4,1)	3.1903	
13	(2,2)	3.5532	
14	(2,2)	3.5584	
15	(0,3)	3.6642	axisymmetric



(a) 1st mode

(b) 2nd mode

(c) 4th mode

(d) 6th mode

(e) 7th mode

(f) 9th mode

(g) 11th mode

(h) 13th mode

(i) 15th mode

Figure 3-5: Eigenmodes of a circular membrane

Table 3-15: Simulation settings frequency response circular membrane

Parameter	Value	Description
ζ	5.0×10^{-4}	Relative damping of all modes
$(x_\mathrm{p}, y_\mathrm{p})$	$R \cdot (0.2, 0.5)$	Coordinates of excitation & measurement point
F_ext	{10,50,100,200} pN	External direct forcing range
M_modes	{1,2,4,6,7,9,11,13,15}	Set of included mode indices

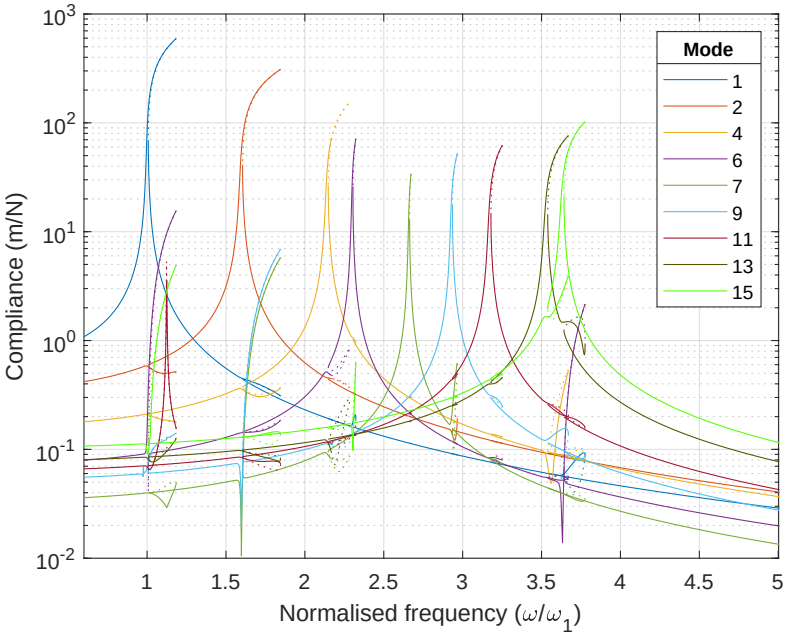


Figure 3-6: Modal frequency response of a circular membrane for external force of 2×10^{-11} N at $(x_p, y_p) = R \cdot (0.2, 0.5)$.

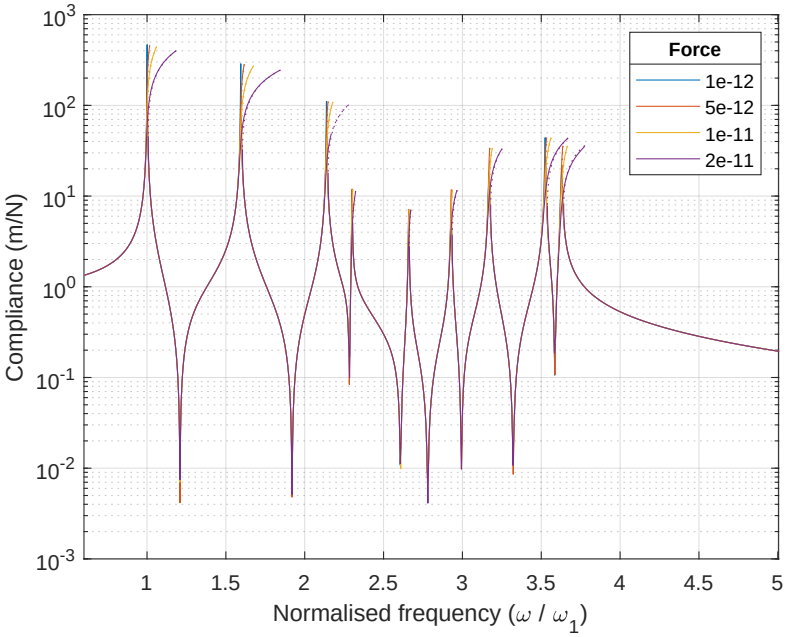


Figure 3-7: Frequency response of a circular membrane for increasing external force level. Excitation and measurement at the same point (collocated), at $(x_p, y_p) = R \cdot (0.2, 0.5)$.

3-1-5 Square membrane

In this section, a square membrane is analysed. For thin structures, bending stiffness can be ignored and almost all the modes are different from the circular membrane. Some eigenmodes are shown in Figure 3-12.

The physical quantities describing it's geometry and physics are listed in Table 3-16. Again, the STEP method is applied using static condensation (so in-plane modes are not explicitly present) for a ROM including the three first non-degenerate eigenmodes. Modal linear coefficients are listed in Table 3-17, and the non-dimensionalising terms for both linear and nonlinear quantities are listed in Table 3-18. Non-dimensionalised cubic stiffness coefficients are shown in Table 3-19 (the quadratic terms appear to all equal zero).

Table 3-16: Properties square membrane

Parameter	Formula	Units	Description
E	-	Pa	Young's modulus
h	-	m	Thickness
L	-	m	Length
ν	-	-	Poisson's ratio
ρ	-	$\mathrm{kg/m^3}$	Density
σ	-	Pa	Stress
$\tilde{\rho}$	ρh	$\mathrm{kg/m^2}$	Mass per unit area
T_0	σh	$\mathrm{N\,m^{-1}}$	Membrane tension

Table 3-17: Non-dimensionalised linear modal values of a square membrane. Index included modes: 1,2,4.

	Eq. 1	Eq. 2	Eq. 3
m/m_t	0.25	0.215	0.25
k/k_0	4.93	10.6	19.7
ω/ω_0	1	1.58	2

Table 3-18: Scaling values square membrane

Linear			Nonlinear	
m_t	k_0	ω_0	a_dim	b_dim
$L^2h\rho$	T_0	$\frac{\pi}{L}\sqrt{\frac{2T_0}{\tilde{\rho}}}$	$[-]$	$\frac{\pi}{1.27-0.97\nu}\frac{Eh}{L^2}$

Table 3-19: Non-dimensionalised cubic coefficients square membrane. Index included modes: 1,2,4.

	Eq. 1	Eq. 2	Eq. 3
b_{111}	3.66	0	−0.000 152
b_{112}	0	11.7	0
b_{113}	−0.000 145	0	15.3
b_{122}	11.7	−0.000 159	10.5
b_{123}	−0.000 31	21	−0.001 44
b_{133}	15.3	−0.000 251	0.001 62
b_{222}	−0.000 21	21.9	0.000 831
b_{223}	10.5	0.000 977	58.4
b_{233}	−0.000 307	58.4	−0.001 59
b_{333}	0.000 484	0	58.5

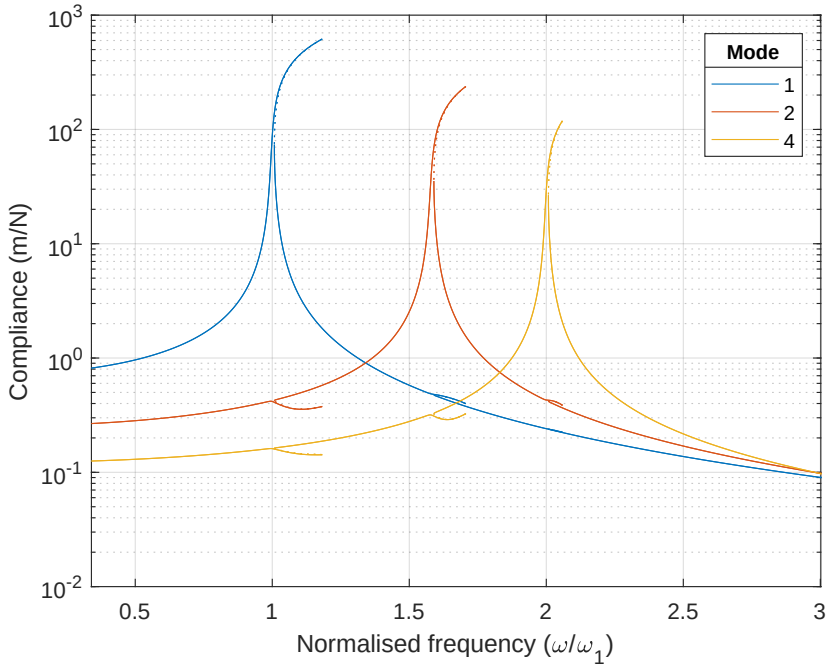


Figure 3-8: Modal frequency response of a square membrane for external force of 2×10^{-11} N at $(x_p, y_p) = R \cdot (0.2, 0.5)$.

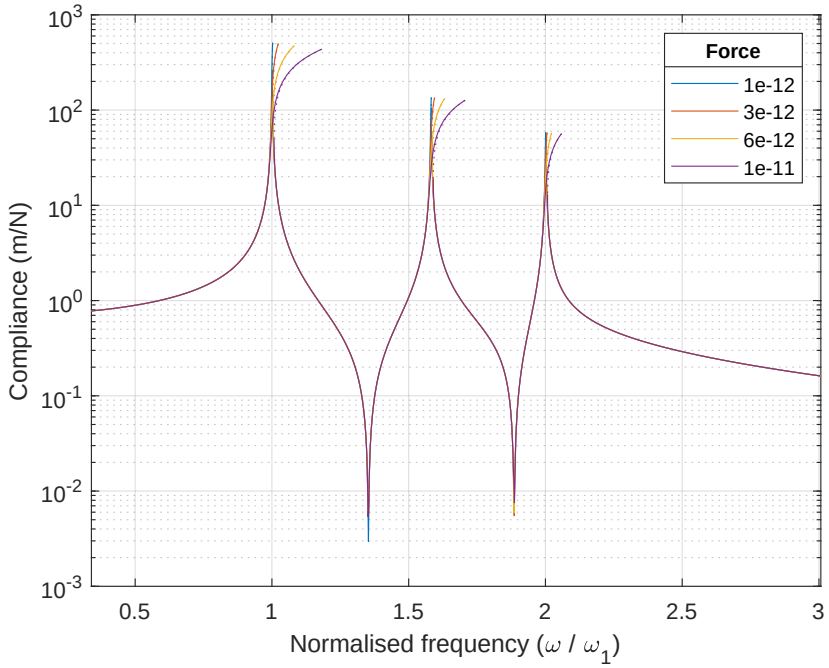


Figure 3-9: Frequency response of a square membrane for increasing external force level. Excitation and measurement at the same point (collocated), at $(x_p, y_p) = L/2 \cdot (0.2, 0.5)$.

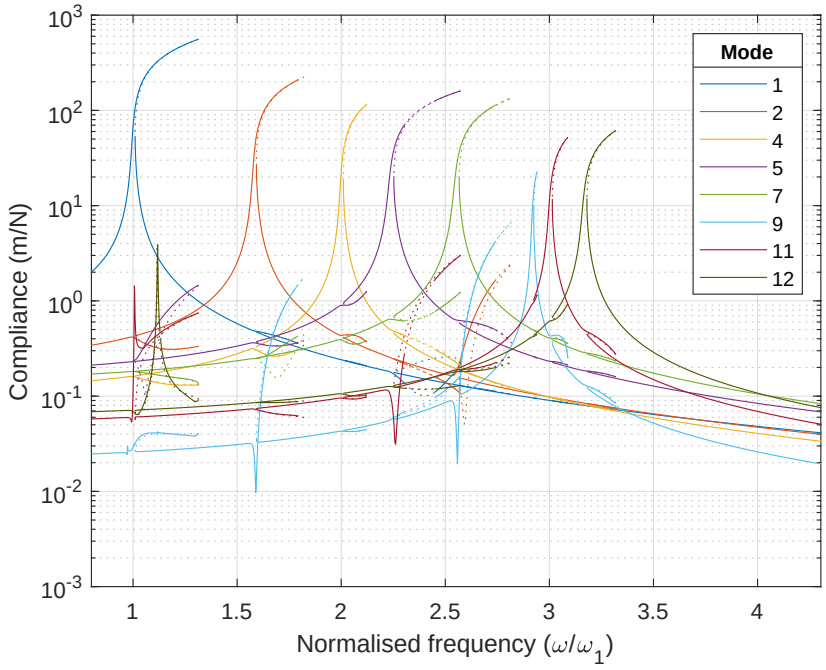


Figure 3-10: Modal frequency response of a square membrane for external force of 2×10^{-11} N at $(x_p, y_p) = R \cdot (0.2, 0.5)$. The ROM includes 8 modes.

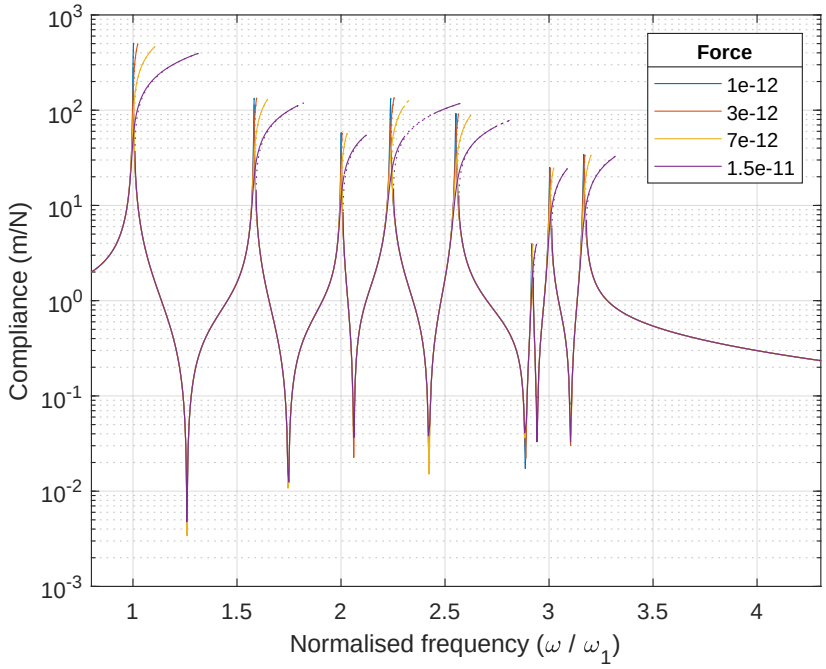


Figure 3-11: Frequency response of a square membrane for increasing external force level. Excitation and measurement at the same point (collocated), at $(x_p, y_p) = L/2 \cdot (0.2, 0.5)$. The ROM includes 8 modes.

Table 3-20: Simulation settings frequency response square membrane

Parameter	Value	Description
ζ	5.0×10^{-4}	Relative damping of all modes
$(x_\mathrm{p}, y_\mathrm{p})$	$L/2 \cdot (0.2, 0.5)$	Coordinates of excitation & measurement point
F_ext	{1,3,7,15} pN	External direct forcing range
M_modes	{1,2,4,5,7,9,11,12}	Set of included mode indices

$$\omega_{n_x n_y} = c\pi\sqrt{\frac{n_x^2}{a^2} + \frac{n_y^2}{b^2}} \tag{3-5}$$

Table 3-21: Ratio of eigenfrequency w.r.t. the fundamental eigenmode $\omega_{n_x n_y}/\omega_{11}$, found by Greiner [14]. The same values yield from modal analysis in Comsol.

$n_y \setminus n_x$	1	2	3	4
1	1.00	1.58	2.24	2.92
2	1.58	2.00	2.55	3.16
3	2.24	2.55	3.00	3.54
4	2.92	3.16	3.54	4.00

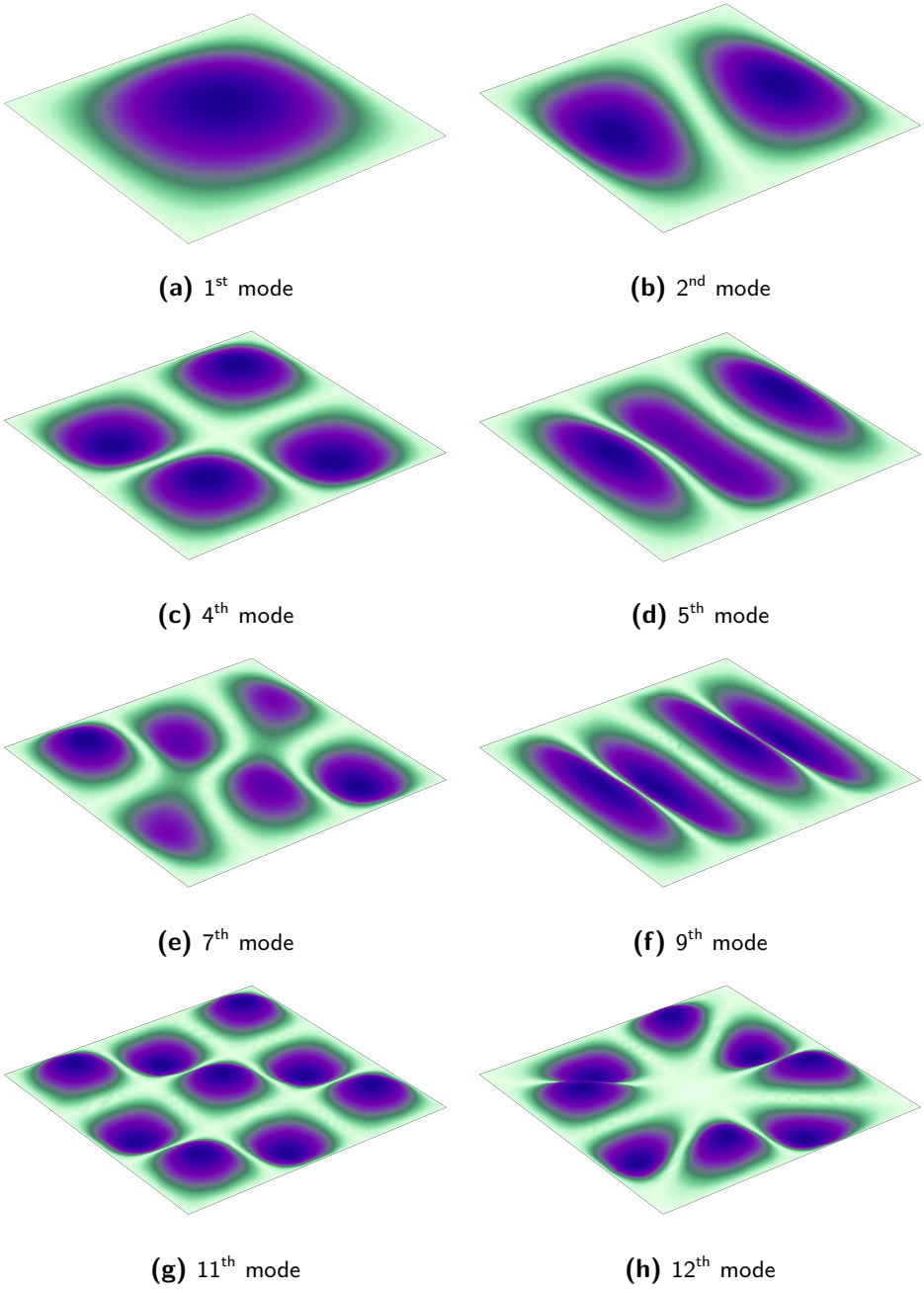
Table 3-22: Normalised eigenfrequencies (of non-degenerate modes) of a square membrane

Index	Waveindex(n_x, n_y)	$\omega_{n_x n_y}/\omega_{11}$
1	(1,1)	1.00
2	(2,1)	1.58
4	(2,2)	2.00
5	(3,1)	2.24
7	(3,2)	2.55
9	(4,1)	3.00
11	(3,3)	2.92
12	(4,2)	3.16

3-1-6 Cantilever

The structures analysed before showed nonlinear stiffness due to stretching of the mid-plane. A simple beam clamped at one end does not show this axial stretching. However, nonlinearities are still present due to the nonlinear kinematic relation [3] & [1].

The physical quantities describing its geometry and physics are listed in Table 3-23. Again, the STEP method is applied using static condensation (so in-plane modes are not explicitly present) for a ROM including the three first non-degenerate eigenmodes. Modal linear coefficients are listed in Table 3-24, and the non-dimensionalising terms for both linear and nonlinear quantities are listed in Table 3-25. Non-dimensionalised cubic stiffness coefficients are shown in Table 3-26 (the quadratic terms appear to all equal zero).



(a) 1st mode **(b)** 2nd mode

(c) 4th mode **(d)** 5th mode

(e) 7th mode **(f)** 9th mode

(g) 11th mode **(h)** 12th mode

Figure 3-12: Eigenmodes of a square membrane

Table 3-23: Properties cantiliver

Parameter	Value	Description
E	210 GPa	Young's modulus
h	3 mm	Thickness
B	4 cm	Width
L	0.5 m	Length
ν	0	Poisson's ratio
ρ	7850 kg/m^3	Density
x_p	L/3	Point of excitation

Table 3-24: Non-dimensionalised linear modal values cantiliver

	Eq. 1	Eq. 2
m/m_t	0.25	0.25
k/k_0	1.03	40.5
ω/ω_0	1	6.27

Table 3-25: Scaling values cantiliver [3] & [15]

Linear			Nonlinear	
m_t	k_0	ω_0	a_dim	b_dim
$LBh\rho$	$\frac{3EI}{L^3}$	$1.875^2\sqrt{\frac{EI}{\rho AL^4}}$	$[-]$	$\frac{EI}{L^5}$

Table 3-26: Non-dimensionalised cubic coefficients cantiliver

	Eq. 1	Eq. 2
b_{111}	3.88	5.05
b_{112}	34.2	176
b_{122}	292	556
b_{222}	308	1.5×10^3

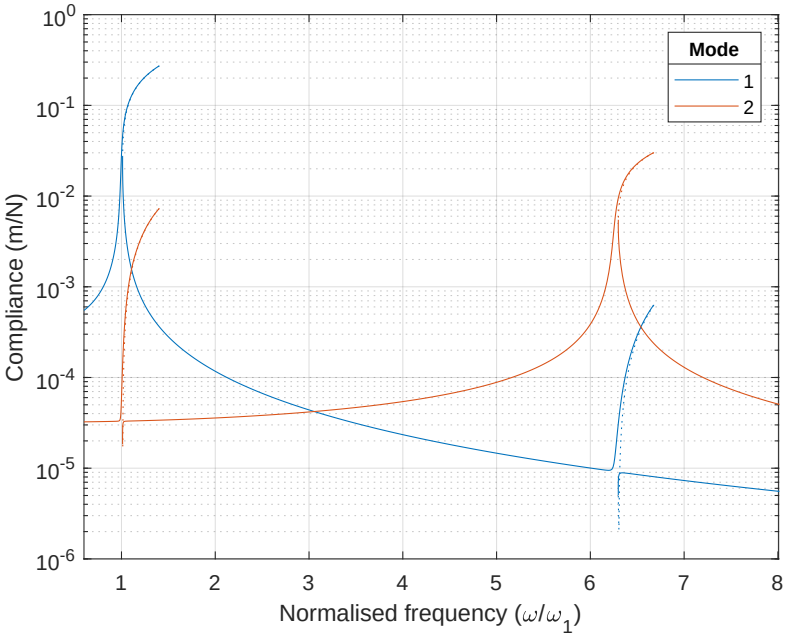


Figure 3-13: Modal frequency response of a cantilever for external force of $2\,\mathrm{N}$.

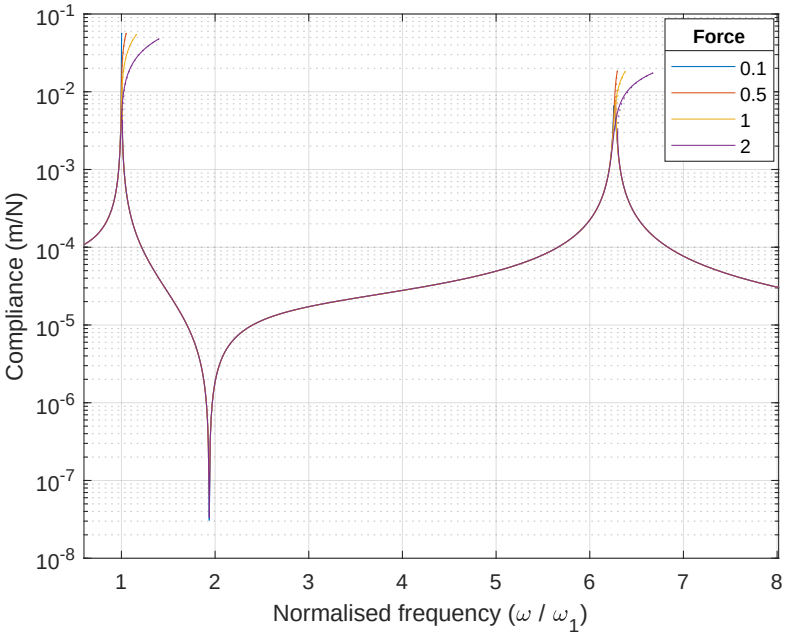


Figure 3-14: Frequency response of a cantilever for increasing external force level. Excitation and measurement at the same point (collocated), at $x_p = L/3$.

Observations

From Table 3-26, it may be observed that the coupling between the two first eigenmodes of a cantilever is quite strong. There are no values negligible, contrary to the string discussed in section 3-1-2. Here, probably the effect of a different source of nonlinearity becomes visible. Further note that because the stiffness increases very fast with mode index, the eigenfrequency sequence is quadratic. Therefore, the eigenfrequency spectrum is way less dense than for a string or membrane, reducing the modal interaction.

3-2 Parametric excitation

Until this point, the frequency response was obtained by direct harmonic forcing, as present in the right-hand-side of (3-1). However, it is also possible to excite the structure parametrically. This implies that the one of the parameters in (3-1) becomes time-dependent. This can be achieved for example if a coefficient is temperature dependent, while temperature is varied over time.

Structures like strings obtain their linear stiffness from pre-tension. When heated up, they will tend to expand. When they are pre-tensioned however, the stress is lowered. As a result, the linear stiffness decreases. Using this property, by varying temperature, stiffness can be varied over time. The equation of motion for a linear system writes:

$$m\ddot{q}(t) + k(t)q(t) = 0 \tag{3-6}$$

When the linear stiffness and direct excitation force are varied harmonically with ω_f, we obtain:

$$m\ddot{q}(t) + k(1 + \gamma \sin(\omega_\mathrm{f} t))q(t) = F \sin(\omega_f t) \tag{3-7}$$

which is in literature known as a forced Matthieu equation. There are several sources describing this equation and it's phenomena ([2] & [16]). Adding cubic nonlinearity, viscous damping and dividing by the mass yields:

$$\ddot{q}_1 + \mu\dot{q}_1 + \omega_1^2(1 + \gamma \sin(\omega_\mathrm{f} t))q_1 + bq_1^3 = \hat{F} \sin(\omega_\mathrm{f} t) \tag{3-8}$$

with

$$\Lambda = \frac{\hat{F}}{2(\omega_\mathrm{f}^2 - \omega_1^2)} \tag{3-9}$$

This equation was also studied by Ramakrishnan & Feeny, 2012 [17], where the following is observed:

- If there is no specific relation between the excitation frequency ω_f and the natural frequency ω_n, the parametric excitation has no effect on the steady-state solution.
- If $\omega_f = \frac{1}{k}\omega_\mathrm{n}$, superharmonic resonance occurs for $k = 1, 2, ...$
- If $\omega_f = k\omega_\mathrm{n}$, subharmonic resonance occurs for $k = 1, 2, ...$

3-2-1 Superharmonic resonance

Superharmonic resonances occur at excitation frequencies below the natural resonance frequency. For the first superharmonic resonance at $\omega = \frac{1}{2}\omega_n$, the following holds [17]:

$$\begin{aligned} q_{\text{peak}} &= \frac{\gamma\Lambda}{2\mu} \\ \omega_{\text{peak}} &= \frac{1}{2}\omega_n + 3b\Lambda^2(1 + \frac{\gamma^2}{32\mu^2}) \end{aligned} \tag{3-10}$$

Therefore, the peak amplitude tends to decrease the further it is away from the natural resonance, because Λ reduces. For the superharmonic resonance, kind of the same behaviour as for the natural resonance is observed:

- The peak amplitude increases with both direct force and parametric excitation amplitude, but decreases with damping.
- Hardening or softening bends the resonance curve for higher amplitude

Superharmonic resonances appear to reshape certain parts of the solution branch, but they do not create new solutions branches. They also appear when no cubic stiffness is present.

3-2-2 Subharmonic resonance

Subharmonic resonances occur at excitation frequencies above the natural resonance frequency. However, their behaviour is somewhat different from the superharmonic resonances. For the first subharmonic resonance at $\omega = 2\omega_n$, the resonance exists if [17]:

$$2\mu \leq \gamma \tag{3-11}$$

The solution shows the same hardening (or softening) behaviour as the natural resonance peak. In contrary to the superharmonic resonance, the solutions are at new branches. A period doubling bifurcation appears at the intersection of the branches. For one subharmonic resonance, two period doubling bifurcations are present, connecting the start and endpoint of the period doubling branch to the base branch. The original or 'base' solution branch is unstable in between the two period doubling bifurcation points, so in practice the period doubling path is followed.

In Figure 3-15 and Figure 3-16, the frequency response is shown. The period doubling bifurcation points of the subharmonic resonances are clearly visible for each eigenmode, enclosing the unstable parts. (Period doubling branches are not drawn, to clearly show the unstable intervals.) Besides, the first and second superharmonic resonances are visible.

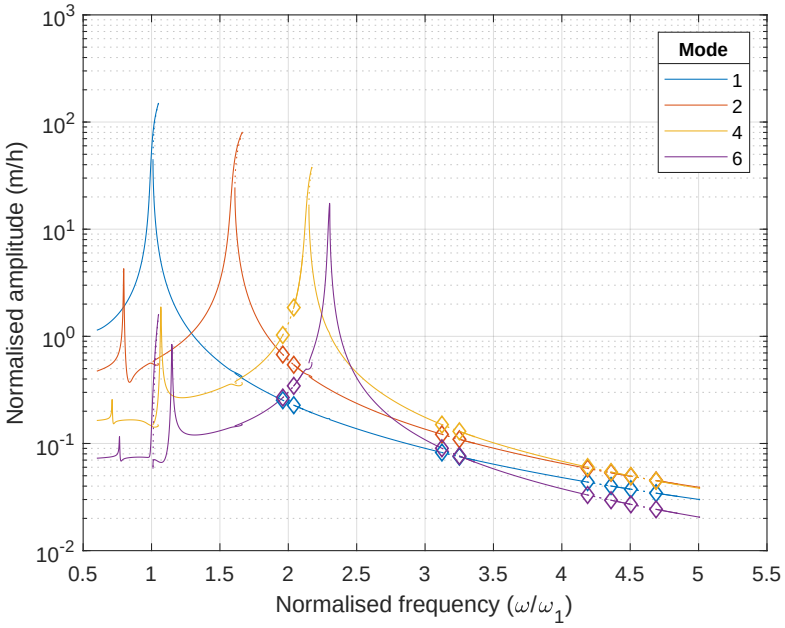


Figure 3-15: Modal frequency response of a circular membrane for external force of $4 \times 10^{-11}\,\mathrm{N}$ at $(x_\mathrm{p}, y_\mathrm{p}) = R \cdot (0.2, 0.5)$ and parametric amplitude f_p of 0.02. Period doubling bifurcations are marked with diamonds. Unstable solutions are indicated with dotted lines.

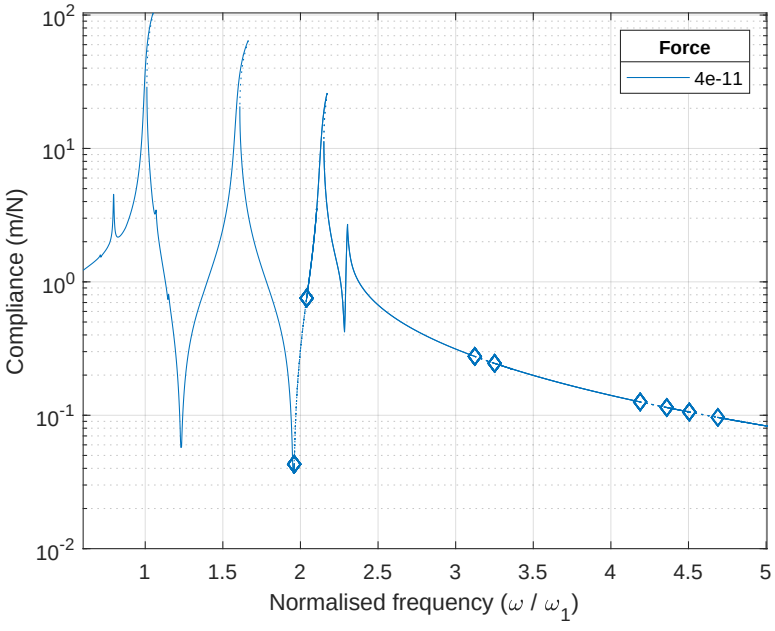


Figure 3-16: Frequency response of a circular membrane for increasing external force level. Excitation and measurement at the same point (collocated), at $(x_p, y_p) = R \cdot (0.2, 0.5)$. Period doubling bifurcations are marked with diamonds. Unstable solutions are indicated with dotted lines.

In Figure 3-17, the frequency response of a parametrically excited single dof system is given. It includes the subharmonic resonance branche, which bends to the right due to cubic stiffness. The excitation levels (both direct force F and parametric excitation level γ) are increased slightly. As visible, the maximum frequency of the parametric resonance branch is very sensitive to the parametric excitaion level γ.

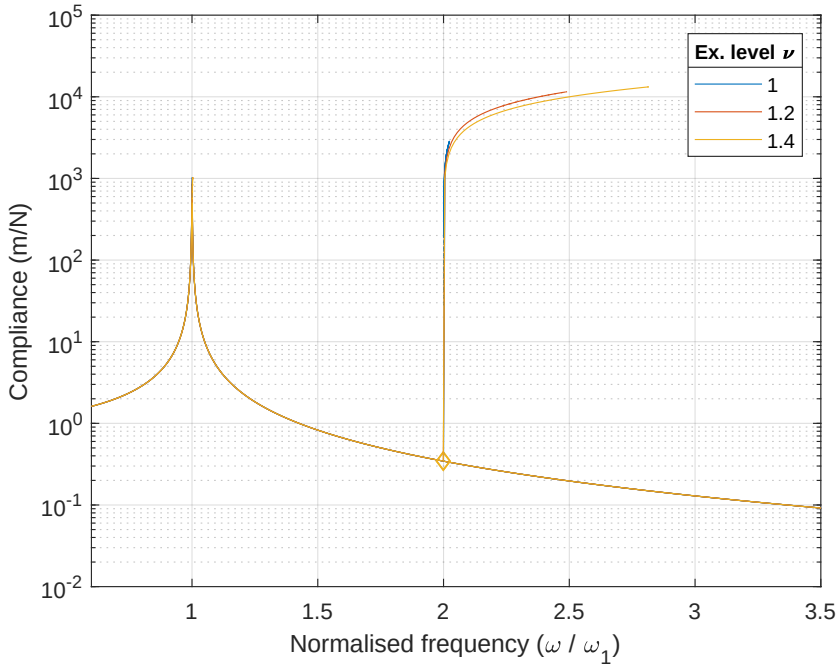


Figure 3-17: Frequency response of a circular membrane for increasing direct (F) and parametric (γ) excitation. Both scale with excitation level ν: $F_i = F_0\nu_i$ and $\gamma_i = \gamma_0\nu_i$, where $F_0 = 1 \times 10^{-12}$,$\gamma_0 = 0.002$. Linear viscous damping with damping ratio $\zeta = 5 \times 10^{-4}$. Excitation and measurement at the same point (collocated), at $(x_p, y_p) = (0, 0)$. Period doubling bifurcations are marked with diamonds. Unstable solutions are indicated with dotted lines.

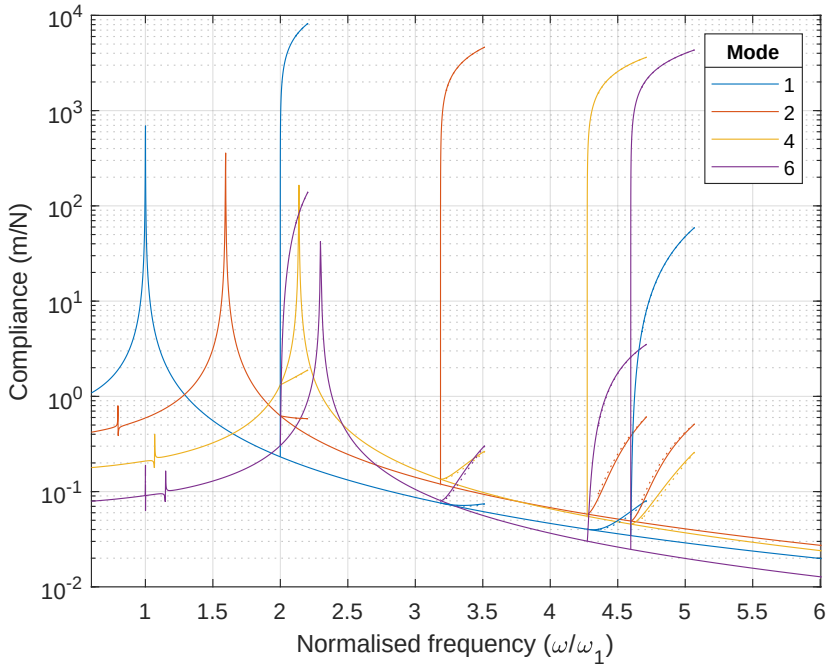


Figure 3-18: Modal frequency response of a circular membrane for increasing direct (F) and parametric (γ) excitation. Parameters: $F = 1.05 \times 10^{-12}$, $\gamma = 2.15 \times 10^{-3}$, $\zeta = 5 \times 10^{-4}$, $(x_p, y_p) = R \cdot (0.2, 0.5)$.

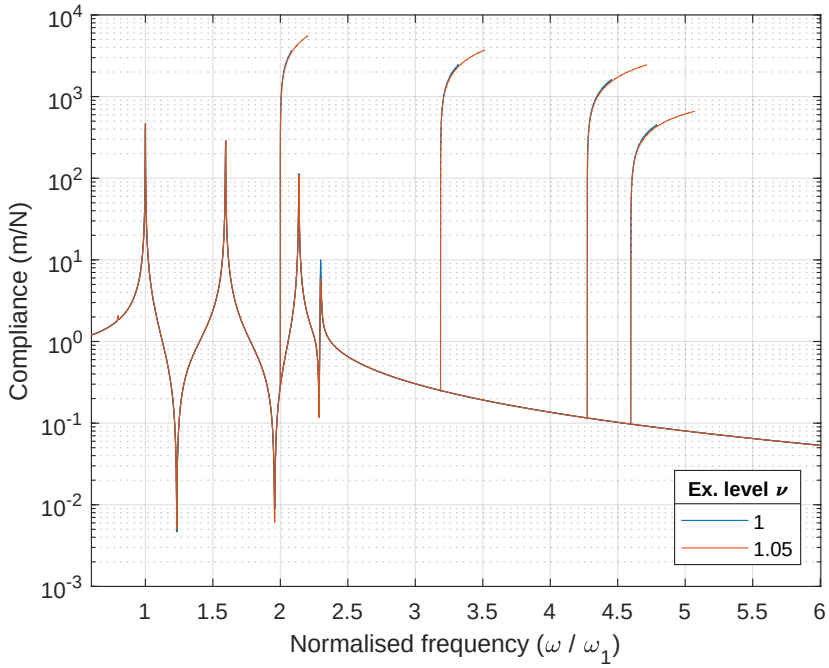


Figure 3-19: Frequency response of a circular membrane for direct (F) and parametric (γ) excitation. ν: $F = F_0\nu$ and $\gamma_i = \gamma_0\nu$, where $F_0 = 1 \times 10^{-12}$,$\gamma_0 = 0.002$. Parameters: $\zeta = 5 \times 10^{-4}$, $(x_p, y_p) = R \cdot (0.2, 0.5)$.

3-2-3 Period doubling bifurcation threshold

From Eq. (3-11), the threshold value for the period doubling bifurcation can be calculated. For the fundamental mode, the result is exact agreement with the simulation shown in Figure 3-19. However, for constant damping ratio ζ, the damping coefficient (c or μ in Eq. (3-11)) is higher for higher modes ($c_i = 2\omega_i\zeta$). Therefore, higher modes should have a higher treshold. However, in the simulation this is not observed, as all parametric resonances appear at the same excitation level. Probably, the assumptions made to derive Eq. (3-11) do not hold for ndof systems.

Proportional damping

In literature [18], it is observed that period doubling branches of higher modes appear at higher parametric excitation levels (Figure 3-20). Therefore, a new simulation is performed using (stiffness) proportional damping, yielding an increasing damping ratio for higher modes. The resulting damping values are listed in Table 3-27.

In Figure 3-21, simulation results are shown. Here, clearly period doubling bifurcations of higher modes pop up at a higher excitation level, similar to experimental data shown in Figure 3-20. Therefore, if the parametric excitation mapping can be assumed to be the same for all modes, it is suspected that the drum experimentally analysed exhibits a similar sort of damping (proportional, or at least having the proportionality $\zeta_i \propto \omega_i$).

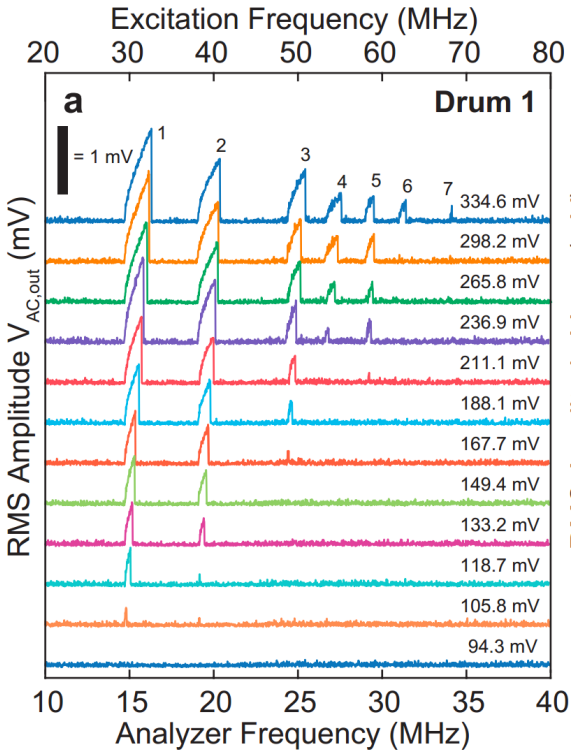


Figure 3-20: Experimental parametric spectrum circular membrane from [18]

Table 3-27: Damping ratio of all modes due to (stiffness) proportional damping. $c_i = \kappa k_i$, $\zeta_i = \kappa\omega_i/2$, $\kappa = 1 \times 10^{-3}$

Mode	ω_i	ζ_i
1	1.00	5.0×10^{-4}
2	1.59	8.0×10^{-4}
4	2.14	10.7×10^{-4}
6	2.30	11.5×10^{-4}

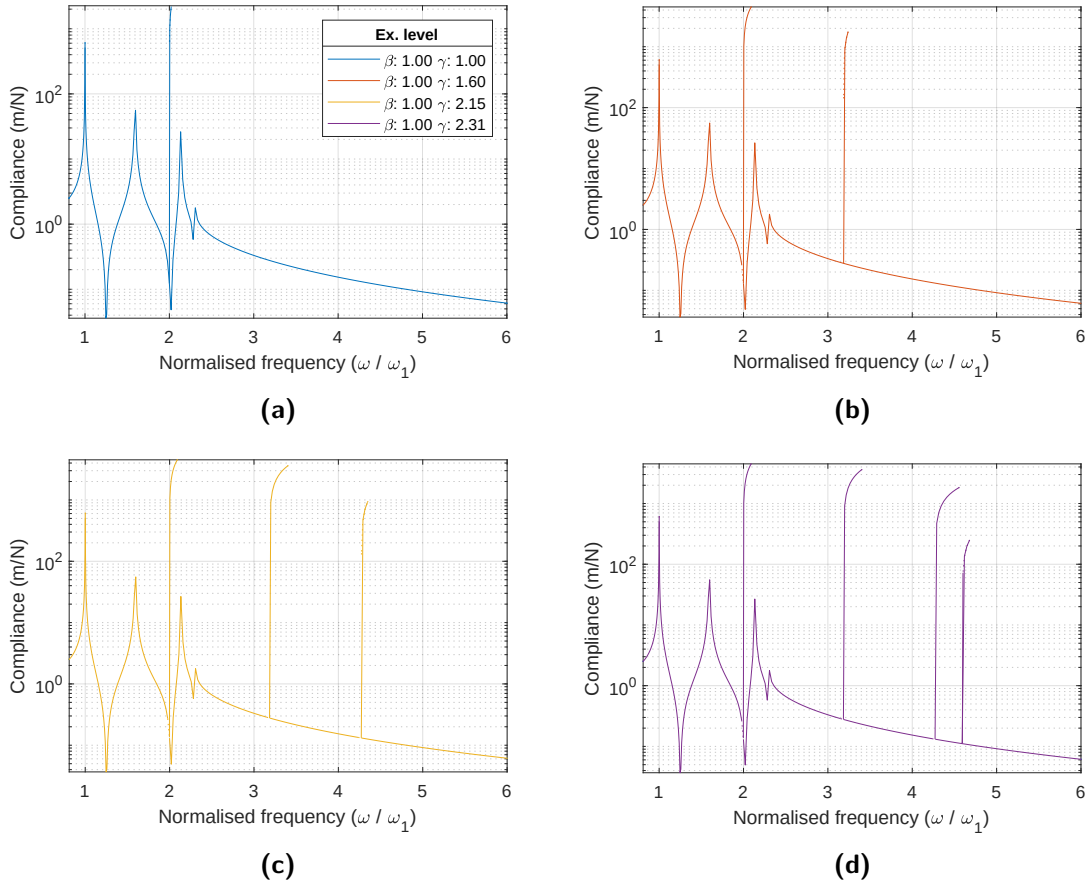


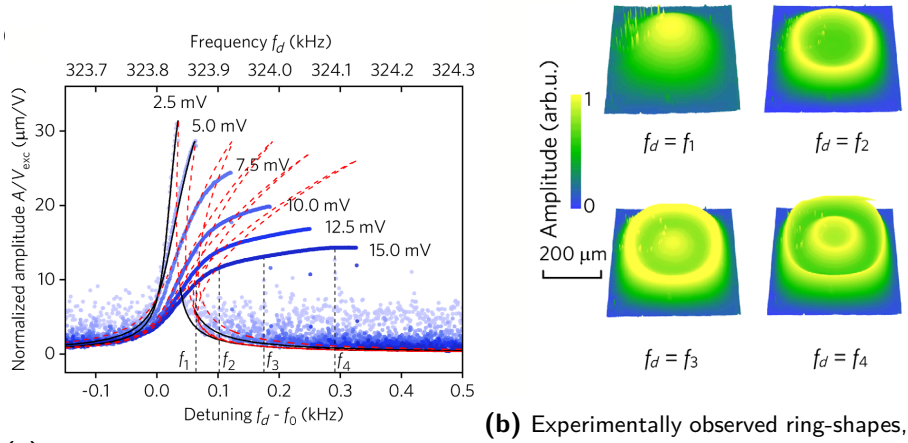
Figure 3-21: Frequency response of a circular membrane with proportional damping for increasing parametric (γ) excitation level. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-12}$,$\gamma_0 = 0.002$. Parameters: $\kappa = 1 \times 10^{-3}$, $(x_p, y_p) = R \cdot (0.18, 0.35)$. Note that only the start of each period doubling resonance branches is shown here. Further note that the excitation levels are chosen to be at the onset (appearance) of each period doubling bifurcation.

3-3 Experimental validation

To show the validity of the proposed methods, in this section simulations are compared to experimental data.

3-3-1 Square membrane including bending stiffness

In literature [19], experiments are performed on a (almost) square membrane, where some special features are observed. At large excitation amplitude, axisymmetric deformation patterns arise (Figure 3-22b) and the mean transmissibility decreases significantly for increasing driving power (Figure 3-22a).



(a) Experimentally obtained frequency response

(b) Experimentally observed ring-shapes, corresponding to driving frequencies shown in sub-figure (a)

Figure 3-22: Experimentally obtained data from [19]

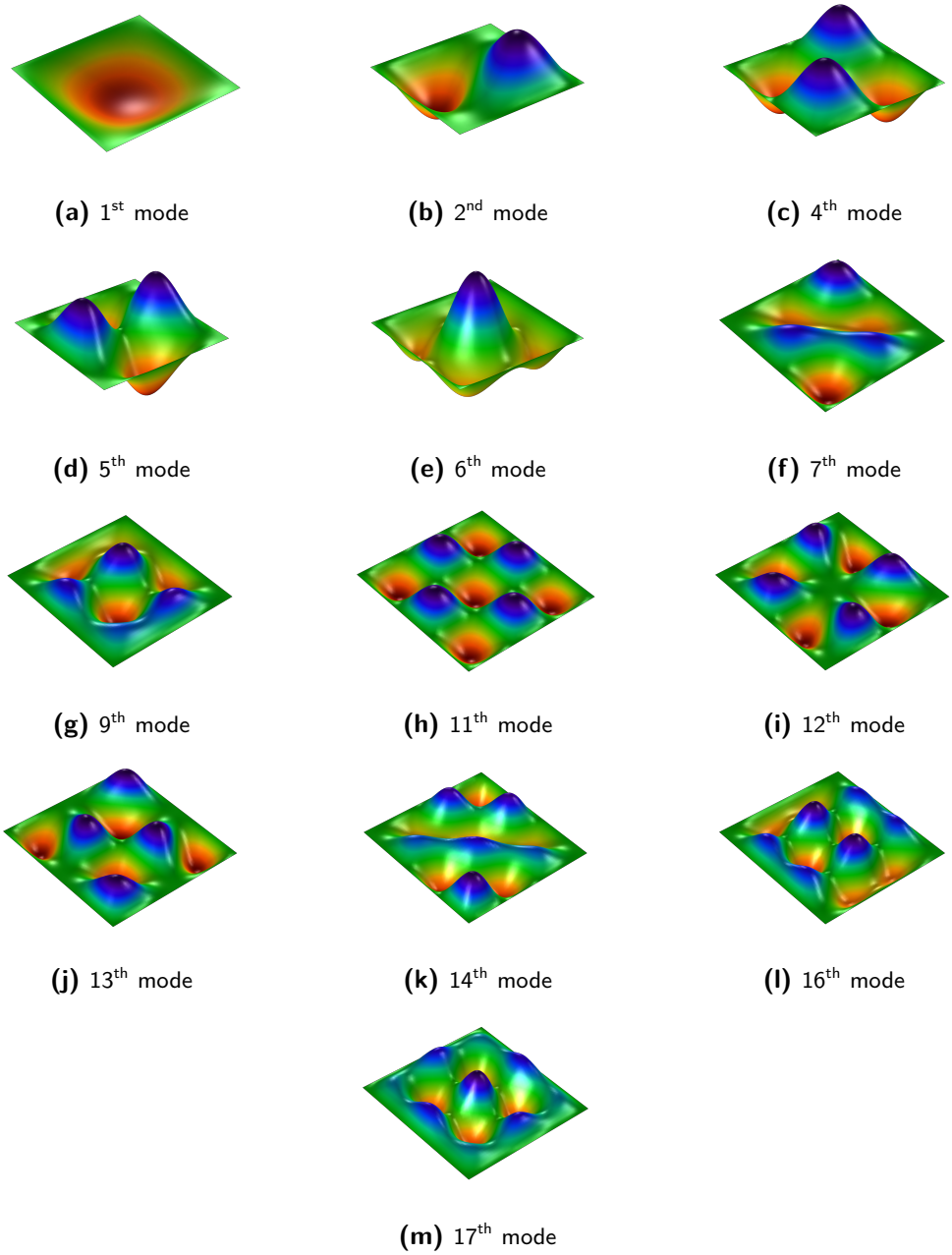
The membrane tested has a length over width ratio slightly larger than one. This causes all eigenmodes (except the fundamental one) to loose anything looking like axisymmetry. However, when including bending stiffness for a square membrane, some eigenmodes become almost axisymmetric. By including those in the ROM, it should be able to describe the observed ring shape.

As mentioned, the average transmissibility decreases less than what would follow from the duffing equation. This could be due to higher modes having strong interaction with the fundamental mode, consuming power. As they are much stiffer, their contribution to the average displacement is low compared to the fundamental mode.

The quantities describing the structure geometry and physics are listed in Table 3-28. Again, the STEP method is applied using static condensation (so in-plane modes are not explicitly present) for a ROM including the 15 first non-degenerate eigenmodes. The eigenmodes are shown in Figure 3-23, and their corresponding frequencies are listed in Table 3-30.

In the experiment, instead of direct forcing, the anchor to which the membrane is mounted is excited, resulting in an anchor diplacement $p(t)$. This yields in a forcing proportional to the mode stiffness (as damping is very low, $\zeta \ll \omega_n$, so velocity-based excitation is neglected):

$$\begin{aligned} \ddot{q}(t) + k(q(t) - p(t)) &= 0 \\ \ddot{q}(t) + kq(t) &= kp(t) \end{aligned} \tag{3-12}$$



(a) 1st mode **(b)** 2nd mode **(c)** 4th mode

(d) 5th mode **(e)** 6th mode **(f)** 7th mode

(g) 9th mode **(h)** 11th mode **(i)** 12th mode

(j) 13th mode **(k)** 14th mode **(l)** 16th mode

(m) 17th mode

Figure 3-23: Eigenmodes of a square plate

Table 3-28: Properties square plate

Parameter	Value	Description
E	240 GPa	Young's modulus
h	478 nm	Thickness
L	413.5 µm	Radius
ν	0.24	Poisson's ratio
ρ	3184 kg/m^3	Density
σ	110 MPa	Pre-stress

Table 3-29: Simulation settings frequency response circular membrane

Parameter	Value	Description
ζ	2.5×10^{-5}	Relative damping of all modes
p_{ext}	$\{3, 6, 12, 20\} \times 10^{-3}$ m	Amplitude base excitation range
M_{modes}	{1,2,4,5,6,7,9,11,12,13,14,16,17}	Set of included mode indices

For harmonic base excitation, the modal frequency response of the square membrane is shown in Figure 3-24, revealing the interaction between the eigenmodes. Figure 3-25a shows the mean amplitude of the whole membrane (which is also reported in [19]). As higher modes increase in amplitude, they contribute to the resulting shape of vibration. The shape at maximum amplitude (indicated with the red dot in Figure 3-25a) is shown in Figure B-3. A cross-section of the shape for different driving frequencies is shown in Figure 3-27.

In Figure 3-25b, the frequency response of the mean amplitude is shown for a model including only the fundamental eigenmode (single degree-of-freedom model). Comparing the 13 dof model (Figure 3-25a) to the sdof model, the following can be observed:

- The 13 dof model has much lower amplitude at the limit point
- The 13 dof model needs higher excitation level (about double) to reach the same limit point frequency

Table 3-30: Normalised eigenfrequencies (of non-degenerate modes) of a square membrane

Index	1	2	4	5	6	7	9	11	12	13	14	16	17
ω_i/ω_0	1.00	1.59	2.01	2.26	2.26	2.58	2.97	3.06	3.23	3.23	3.64	3.71	3.71

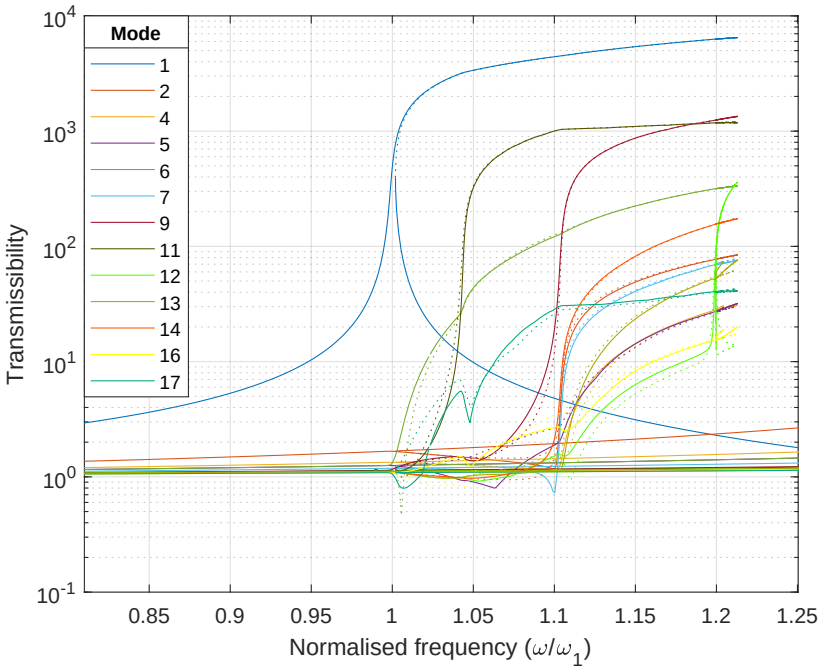
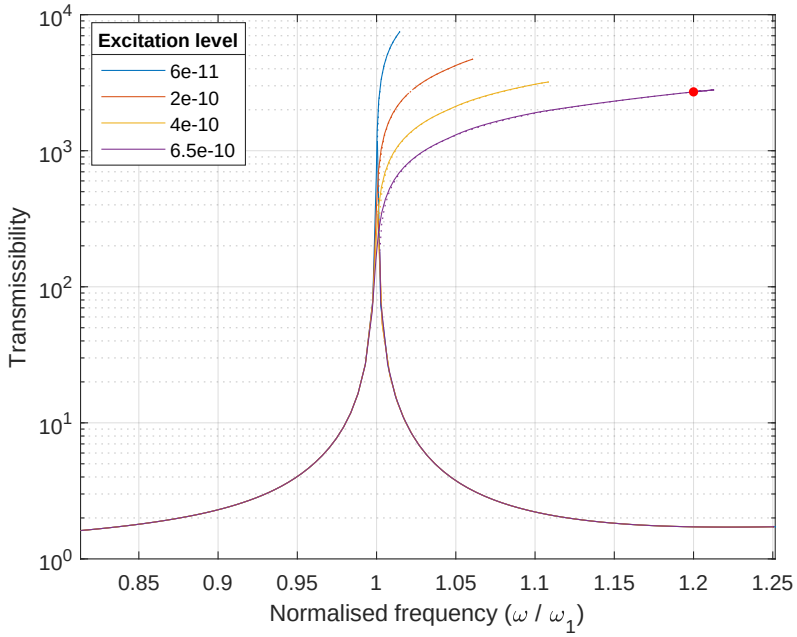
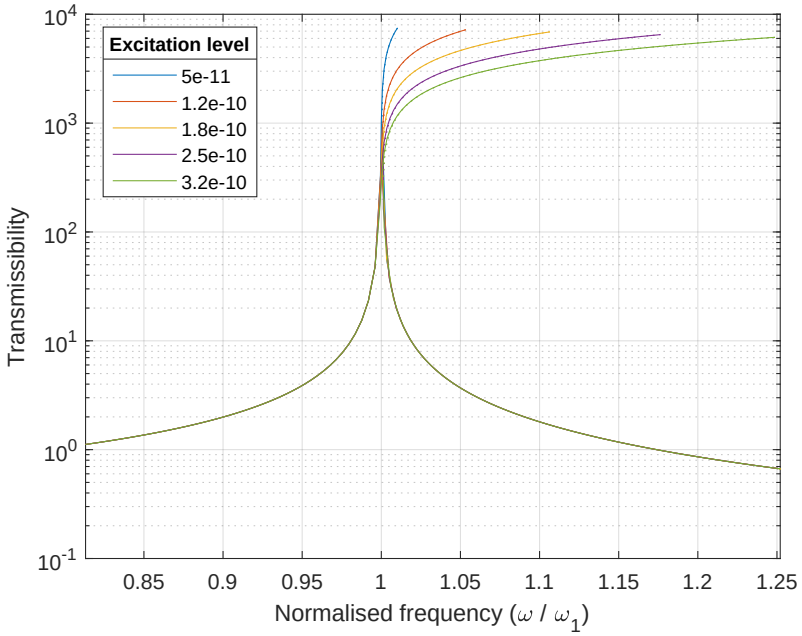


Figure 3-24: Modal frequency response of a square membrane for base excitation of $700\,\mathrm{nm}$. The ROM includes 13 modes.



(a) Frequency response of a square membrane for increasing base excitation level (vibration amplitude is integrated over complete surface). The frequency content at the red dot is shown in Figure 3-26 and Figure B-2.



(b) Frequency response of a square membrane for increasing base excitation level (vibration amplitude is integrated over complete surface). The ROM contains **only the fundamental mode**.

Figure 3-25: Frequency response of a square membrane (including bending stiffness) for increasing base excitation level.

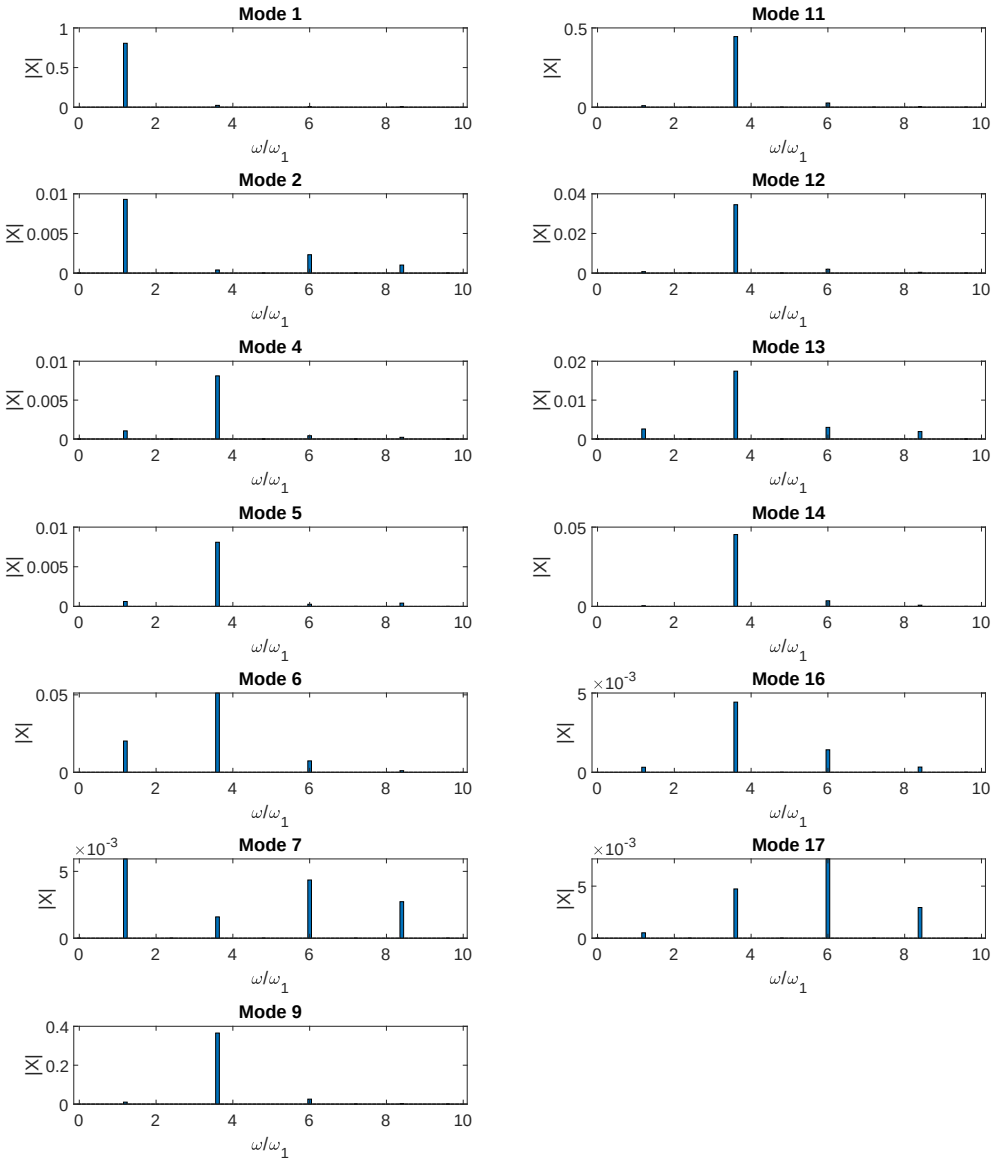


Figure 3-26: FFT of modal signal of stable solution along the resonance curve at $\omega_f = 1.2\omega_1$ (the red dot in Figure 3-25a)

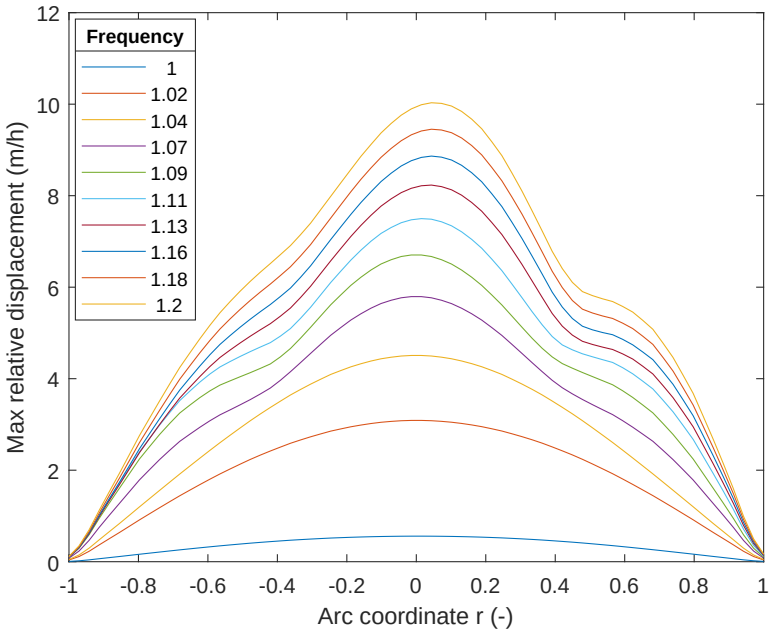


Figure 3-27: Vibration amplitude along cross section $x = 0$ for different points along the resonance peak.

Note that, as analysed in Figure 3-26, most higher modes vibrate at three times the excitation frequency. However, summing the amplitude of vibration of every point still provides a kind of total deflection pattern.

Conclusion

Comparing the results to the experimental results reported in [19], it can be concluded that

- Modal interaction could cause flattening of the resonance curve (Figure 3-25). This can be seen as a way of mode-coupling damping in forced vibration, as energy flows from the fundamental mode to the higher modes, where it is dissipated. In literature [20], observations during ringdown experiments related to this phenomena are reported.
- The rise of ring-shape deformation patterns shows some qualitatively similar behaviour as in the experiment, but cannot completely be explained by nonlinear mode coupling (using linear eigenmodes). Probably other effects like dynamic interaction with the anchor also play a role here.

3-3-2 Circular Graphene membrane

A. Keşkekler has performed experiments on a graphene membrane fabricated on a chip. By illumination with lasers, the membrane is heated up, inducing thermal expansion. As a consequence, the pre-stress and thus the linear stiffness is changed, resulting in parametric

excitation. Additionally, the lasers cause direct forcing, through a photonic force or due to imperfections (which is rather complicated). Because the membrane is very small (it has a diameter of around 5 micron, and thickness in the nanometer range), it's properties are not precisely known. Besides, during fabrication imperfections like wrinkles and tension variations could occur. Moreover, the point of excitation and measurement are only approximately known and the direct and parametric forcing levels have to be estimated from the laser power levels. Nevertheless, as the experiments are performed at different power levels, a lot of information can be gathered from the datasets. The following steps are taken to gather all information:

1. From the linear frequency response, the (first 10) eigenfrequencies of the circular membrane are found (shown in Table 3-33 and Figure 3-28). Despite the asymmetry (inhomogeneous pretension, as listed in Table 3-31), the eigenmodes 4 & 5 and 7 & 8 remain degenerate, resulting in only one resonance peak per set. Mode 9 remains invisible, probably due to the actuation or sensor location.

2. A FEM model is build of a circular membrane clamped at the edge. Plate elements are used, so bending stiffness can be included.

3. From the experimentally found eigenfrequencies, pre-stress (in x -and y direction), bending stiffness and thickness are estimated (listed in Table 3-31). This yields the linear response shown in Figure 3-29b and Figure 3-30b.

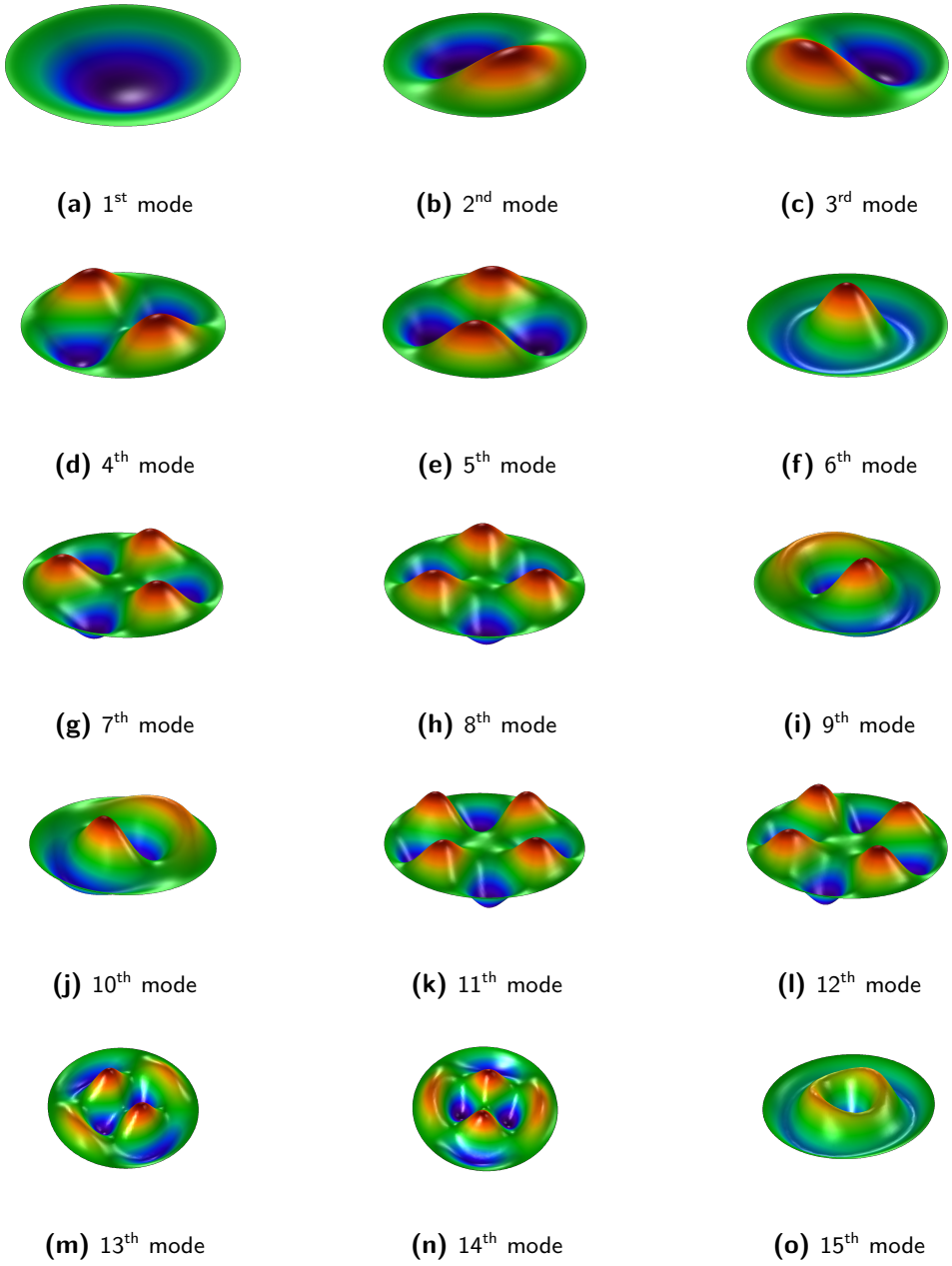
4. From the frequency response in the linear domain, the damping can be estimated (using the half-width bandwidth or the resonance-stiffness ratio).

5. The fundamental eigenmode exhibits hardening, shifting the frequency at maximum amplitude upwards. From this shift, the direct forcing level can be determined.

6. For each higher mode, the modal force can be determined from the hardening. Subsequently, the mapping of the direct force can be determined.

7. From the amplitude at resonance in the linear regime, the sensor-mapping can be determined for each mode. Here, the sensing is assumed to remain linear at higher amplitude.

8. Also the period doubling branch shows hardening. From this, the parametric excitation level can be determined (just like the direct force was extracted).



(a) 1st mode **(b)** 2nd mode **(c)** 3rd mode

(d) 4th mode **(e)** 5th mode **(f)** 6th mode

(g) 7th mode **(h)** 8th mode **(i)** 9th mode

(j) 10th mode **(k)** 11th mode **(l)** 12th mode

(m) 13th mode **(n)** 14th mode **(o)** 15th mode

Figure 3-28: Eigenmodes of Ata's membrane

Table 3-31: Properties circular membrane

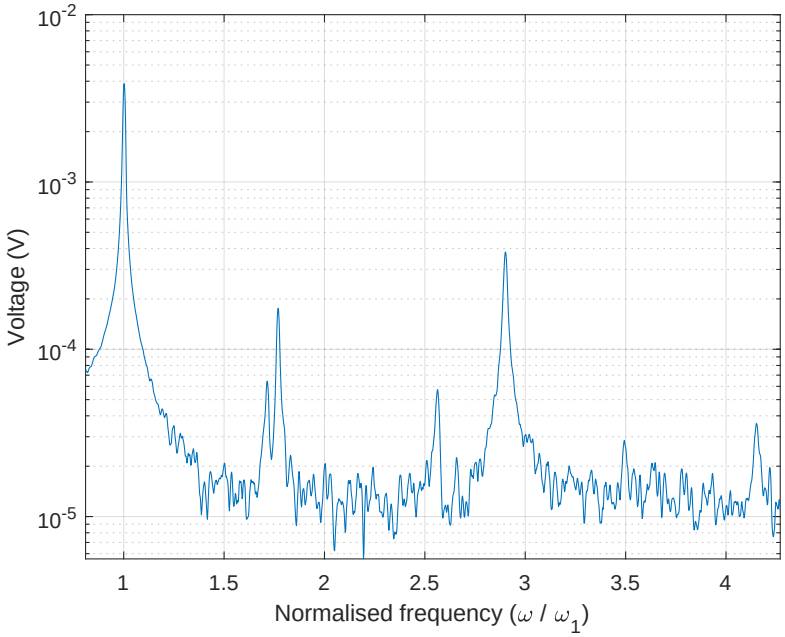
Parameter	Value	Description
E	430 GPa	Young's modulus
h	11.6 nm	Thickness
R	2.5 µm	Radius
ν	0.16	Poisson's ratio
ρ	$2267\,\mathrm{kg/m^3}$	Density
T_{0_x}	$0.321\,\mathrm{N\,m^{-1}}$	Membrane tension in x-direction
T_{0_y}	$0.257\,\mathrm{N\,m^{-1}}$	Membrane tension in y-direction

Table 3-32: Simulation settings frequency response circular membrane

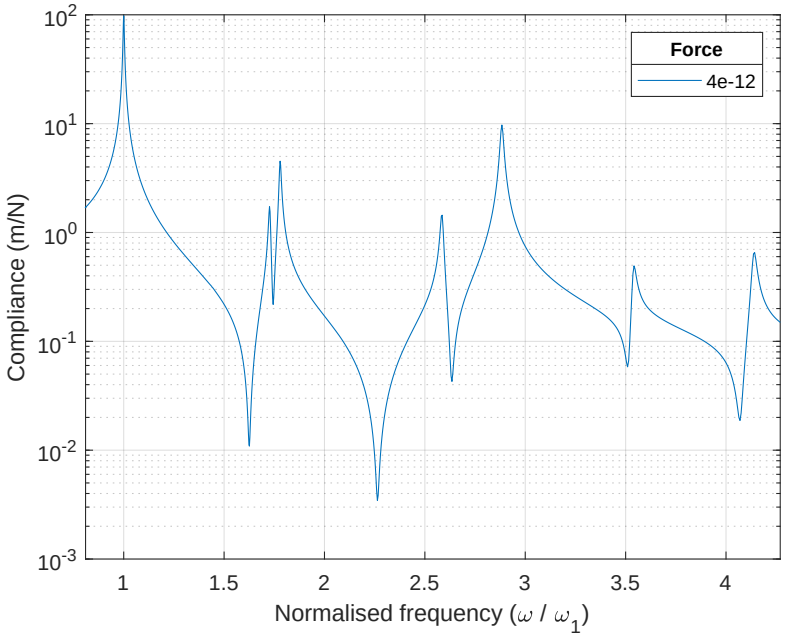
Parameter	Value	Description
ζ	0.0028	Relative damping of all modes
Q	180	Quality factor
F_0	10 pN	External direct forcing multiplier
γ_0	0.01	Parametric excitation multiplier
M_{modes}	{1,2,3,4,6,7,10}	Set of included mode indices

Table 3-33: Normalised eigenfrequencies of a circular membrane with different pre-stress in x and y direction. Note that only the degenerate modes 2 & 3 and 9 & 10 diverge.

Index	Waveindex(t,r)	ω_i/ω_0	Shape
1	(0,1)	1	axisymmetric
2	(1,1)	1.728	
3	(1,1)	1.779	
4	(2,1)	2.586	
5	(2,1)	2.593	
6	(0,2)	2.884	axisymmetric
7	(3,1)	3.540	
8	(3,1)	3.541	
9	(1,2)	4.069	
10	(1,2)	4.140	
11	(4,1)	4.610	
12	(4,1)	4.611	
13	(2,2)	5.447	
14	(2,2)	5.447	
15	(0,3)	5.801	axisymmetric

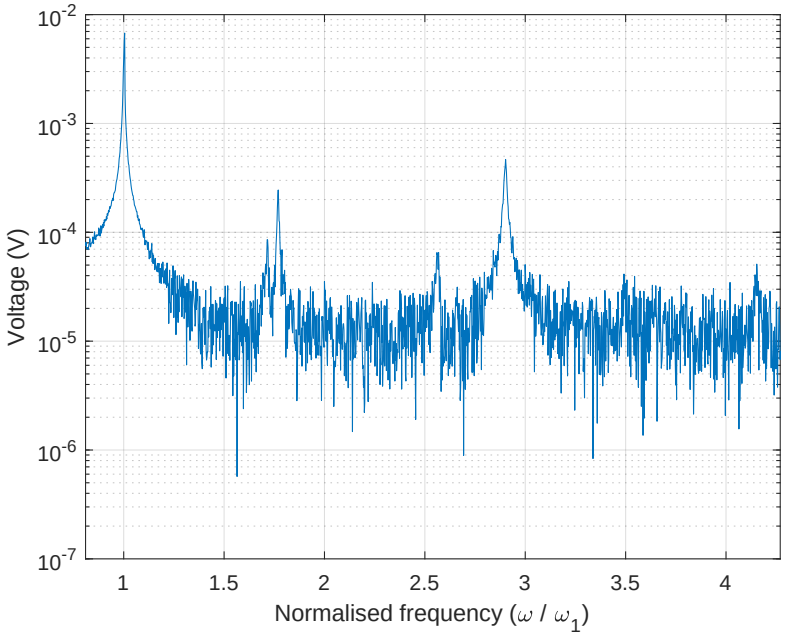


(a) Experimentally obtained frequency response of a membrane (smoothed data).

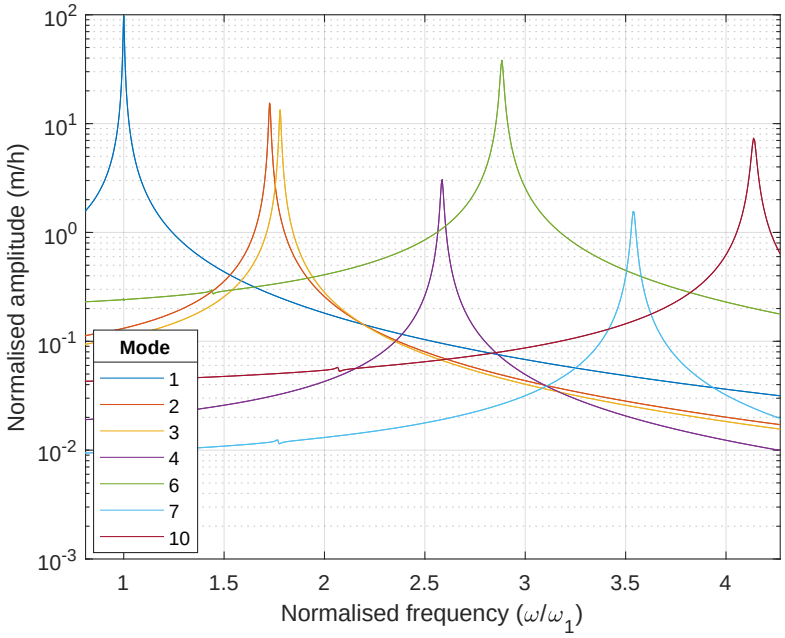


(b) Simulated frequency response of a membrane.

Figure 3-30: Frequency response of a clamped membrane in the linear regime.



(a) Experimentally obtained frequency response of a membrane.



(b) Simulated modal frequency response of a membrane.

Figure 3-29: Frequency response of a clamped membrane in the linear regime.

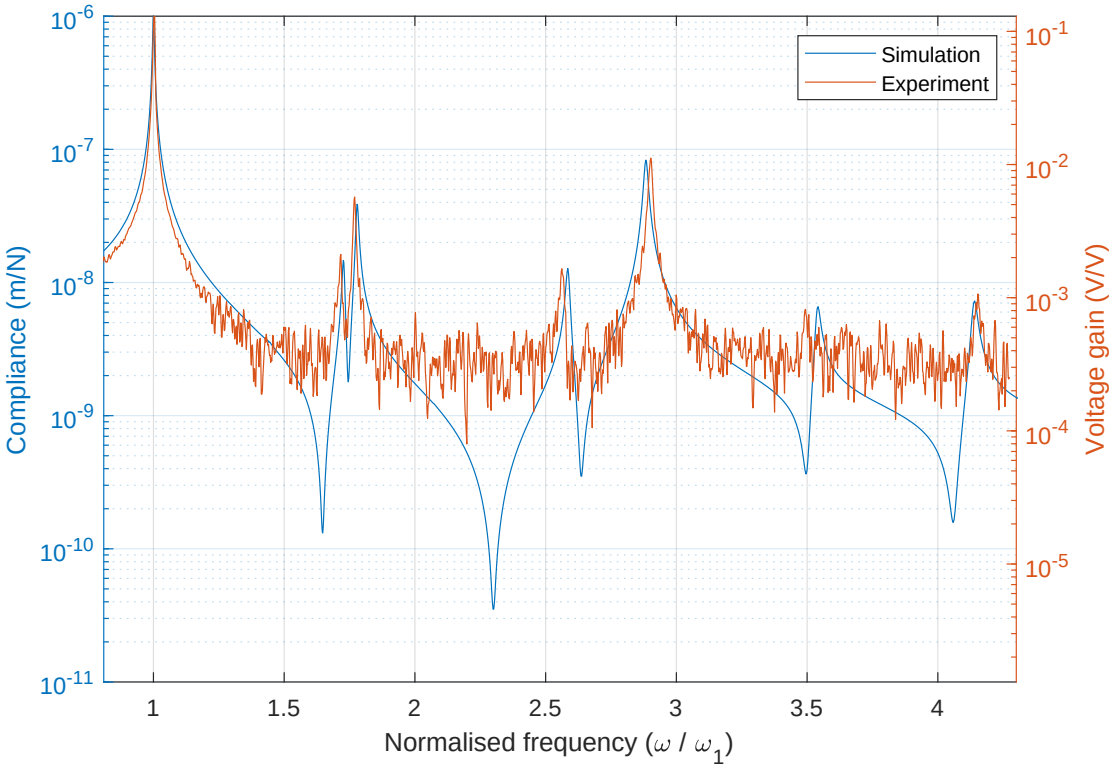
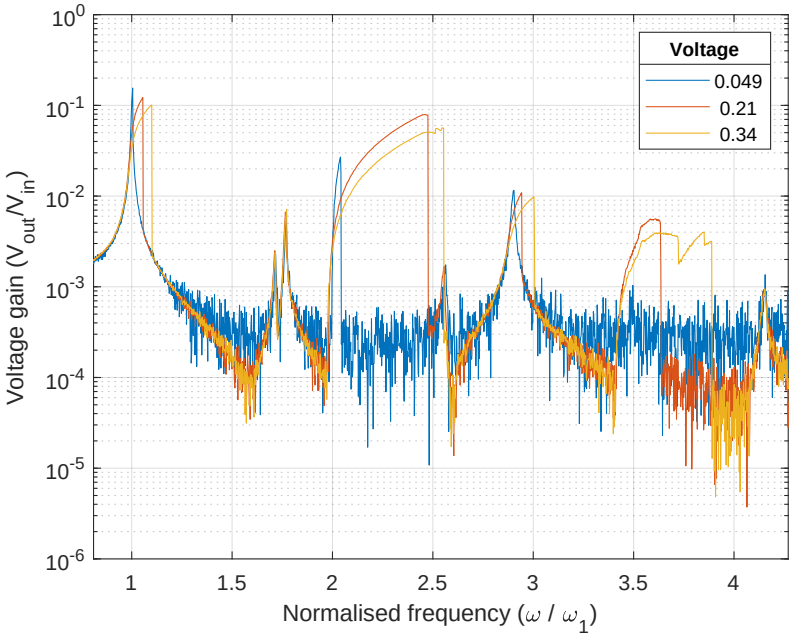
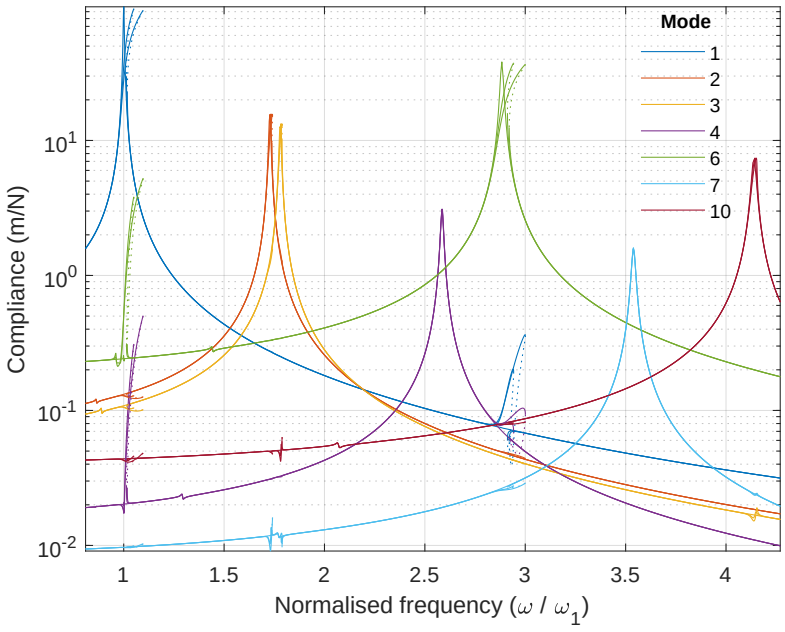


Figure 3-31: Simulation and experiment (smoothed) of a clamped membrane in the linear regime.

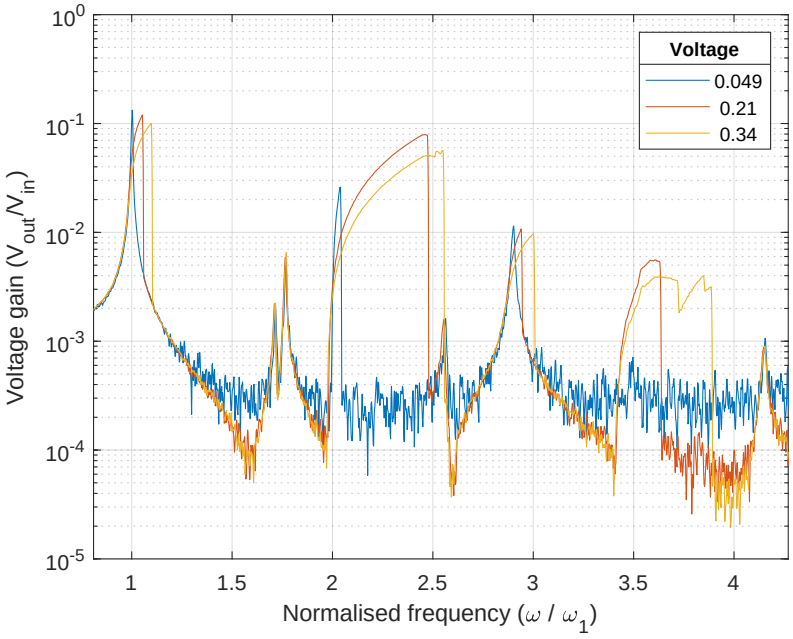


(a) Experimentally obtained frequency response.

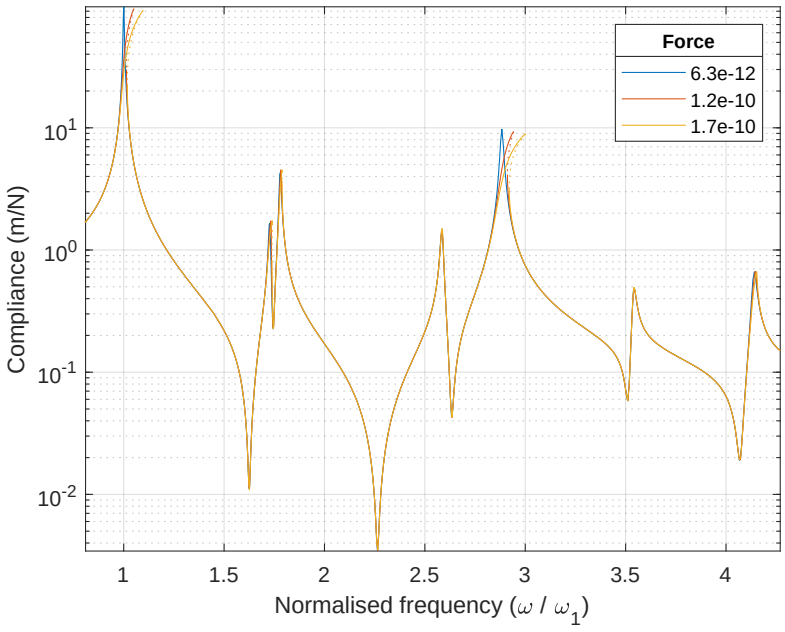


(b) Simulated modal frequency response due to direct forcing.

Figure 3-32: Frequency response of a clamped membrane showing hardening.

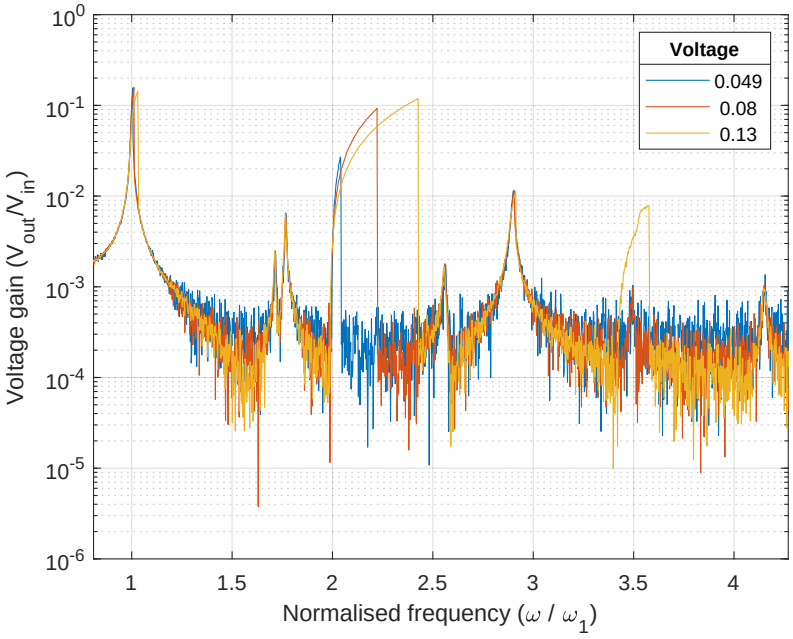


(a) Experimentally obtained frequency response of a membrane (smoothed data).

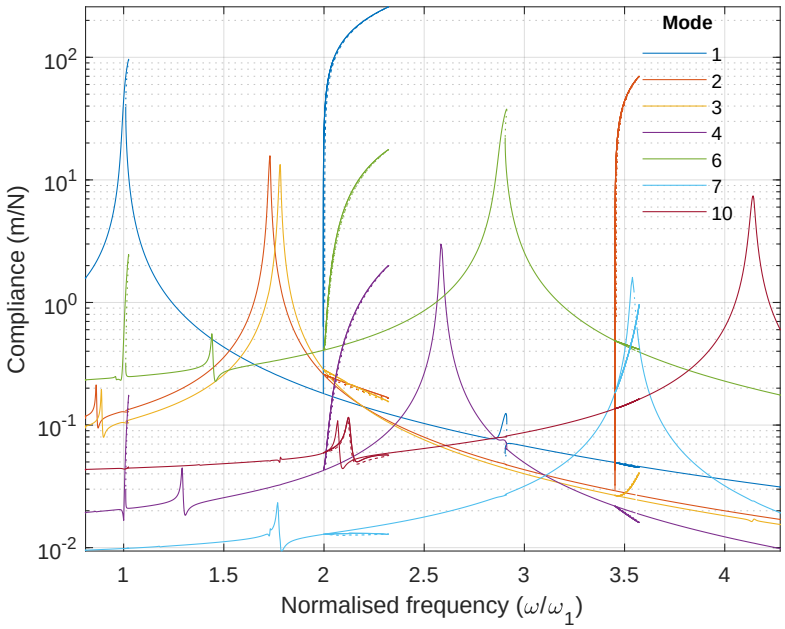


(b) Simulated frequency response of a membrane.

Figure 3-33: Frequency response of a clamped membrane showing hardening.

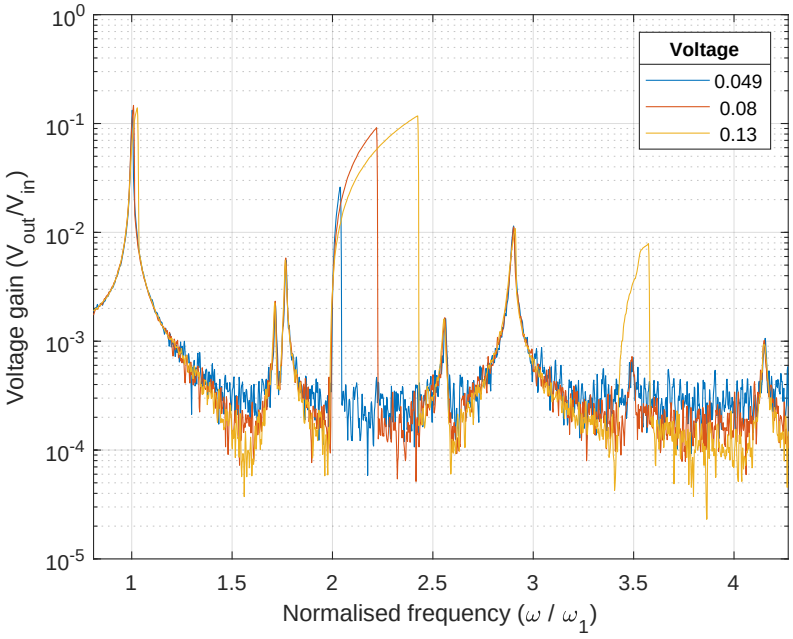


(a) Experimentally obtained frequency response.

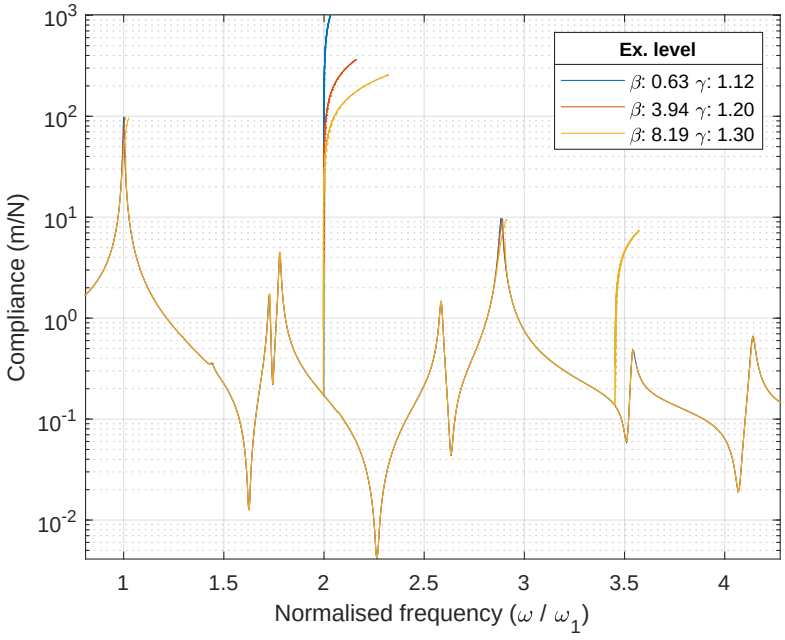


(b) Simulated modal frequency response due to direct and parametric excitation.

Figure 3-34: Frequency response of a parametrically excited clamped membrane showing hardening.



(a) Experimentally obtained frequency response of a membrane (smoothed data).



(b) Simulated frequency response of a membrane due to direct and parametric excitation.

Figure 3-35: Frequency response of a clamped membrane showing hardening.

Conclusion

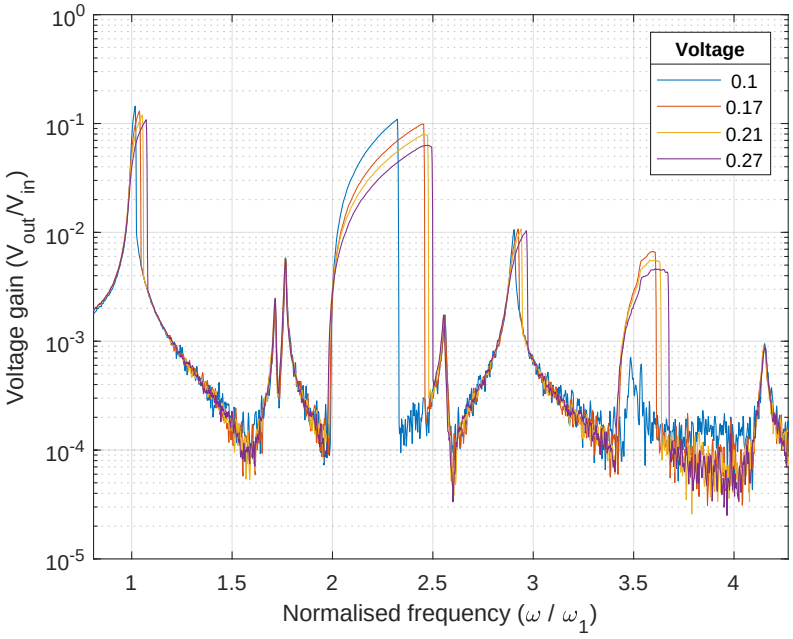
In conclusion, it can be stated that using the Condensed STEP method for a pre-stressed plate, simulations resemble experimental data very well. Both hardening caused by cubic stiffness,

as well as parametric resonance appear in simulations, while only structural parameters like thickness and elasticity (which are only approximately known) are fitted.

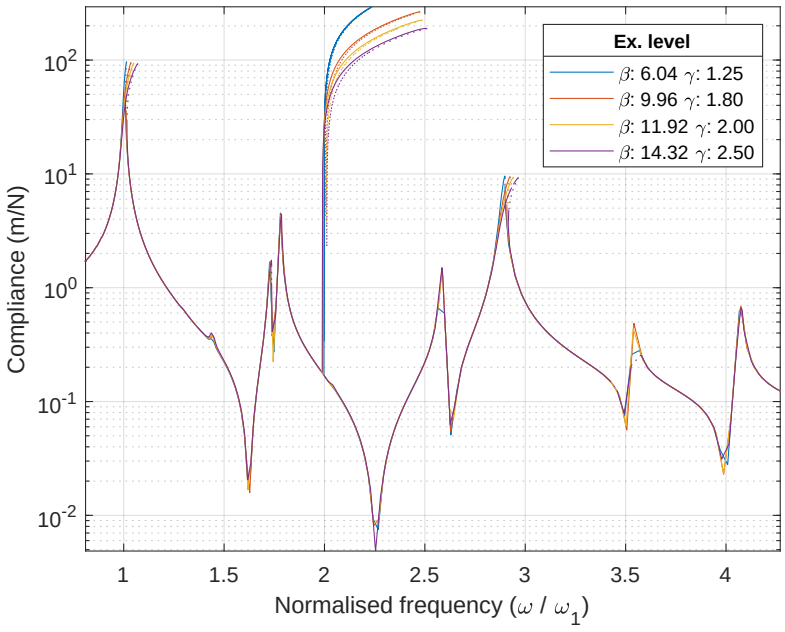
3-3-3 Shooting - internal resonance

When further increasing the excitation level, something strange happens. The limit point of the period doubling branch of the first mode does not increase in frequency anymore (locking, as visible in Figure 3-36a at $\omega/\omega_1 = 2.5$). Then, after a certain threshold is reached, the limit point jumps to a frequency far away, ignoring the natural resonance peaks of mode 4 and 6.

This behaviour was observed in experiments by ir. A. Keşkekler (Figure 3-36a and Figure 3-38a), and shows up in simulations almost identically (Figure 3-37b and Figure 3-38b). Additionally, in Figure 3-37 and Figure 3-39, the modal amplitude of each mode is shown, which reveals the underlying mechanisms. The period doubling resonance peak of the second eigenmode is not simulated, as this is not at our interest at this point.

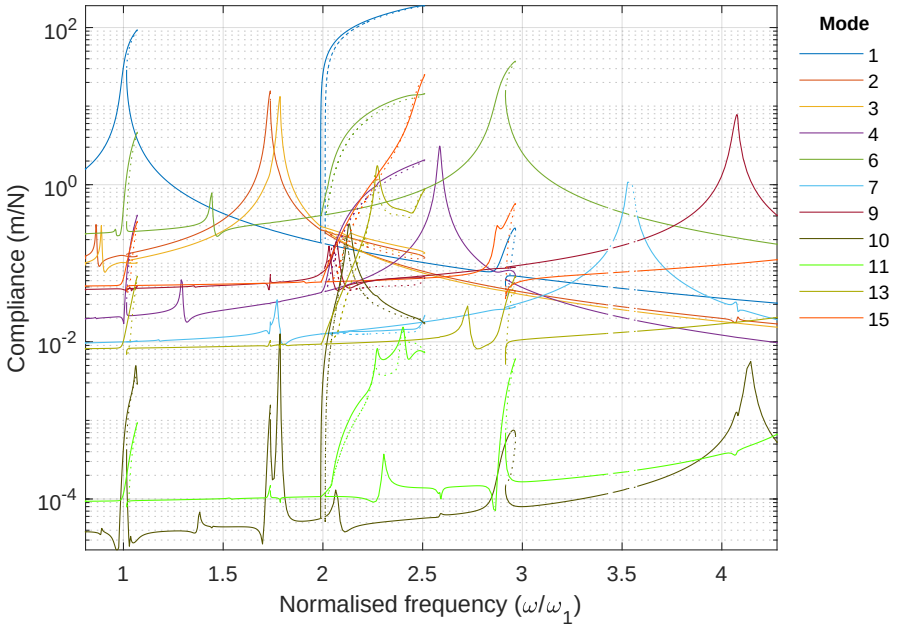


(a) Experimentally obtained frequency response (smoothed data).

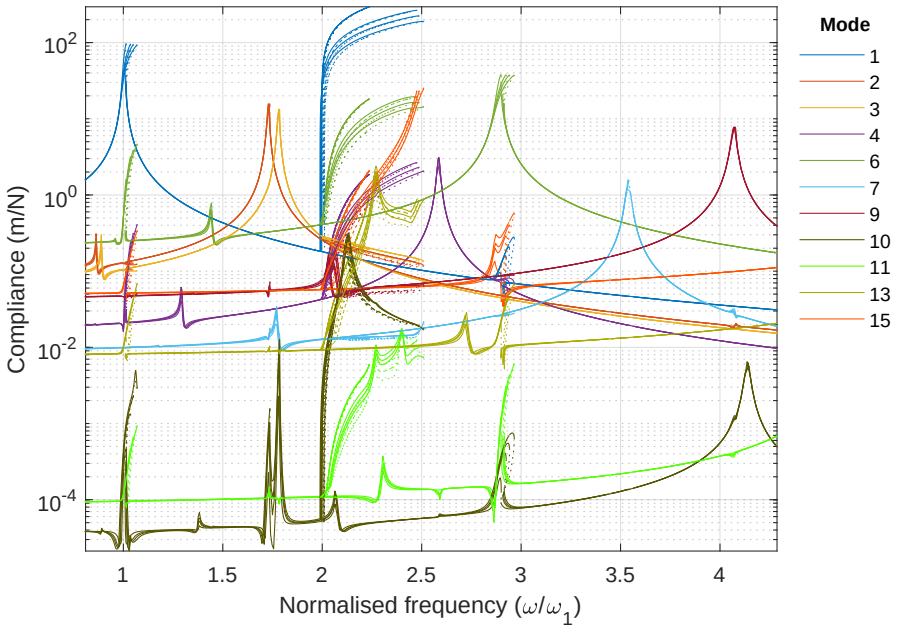


(b) Simulated frequency response for increasing direct (F) and parametric (γ) excitation. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-11}$,$\gamma_0 = 0.01$.

Figure 3-36: Frequency response of a clamped membrane showing locking.

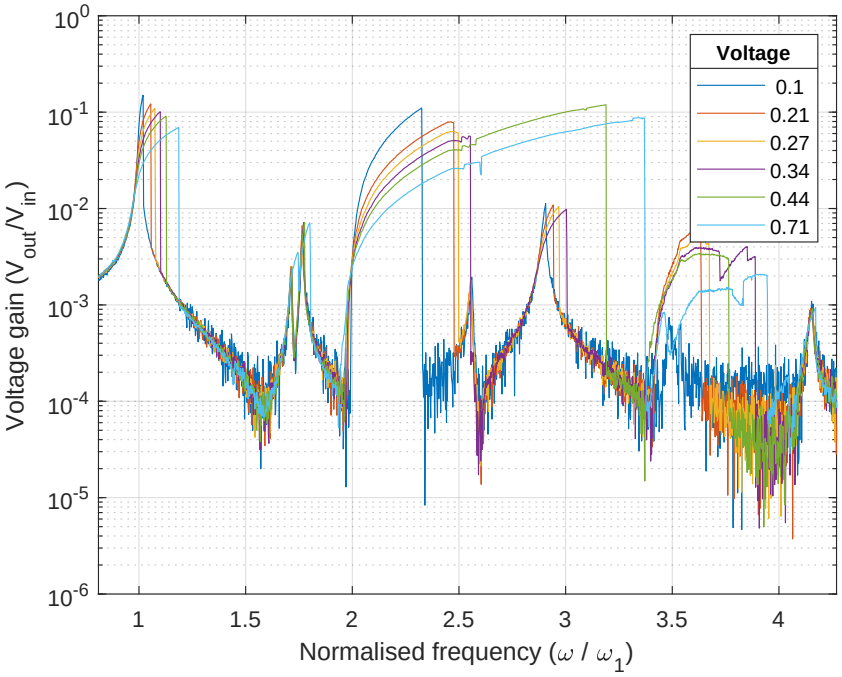


(a) Simulated modal frequency response for excitation levels $\beta = 14.32$ and $\gamma = 2.5$.

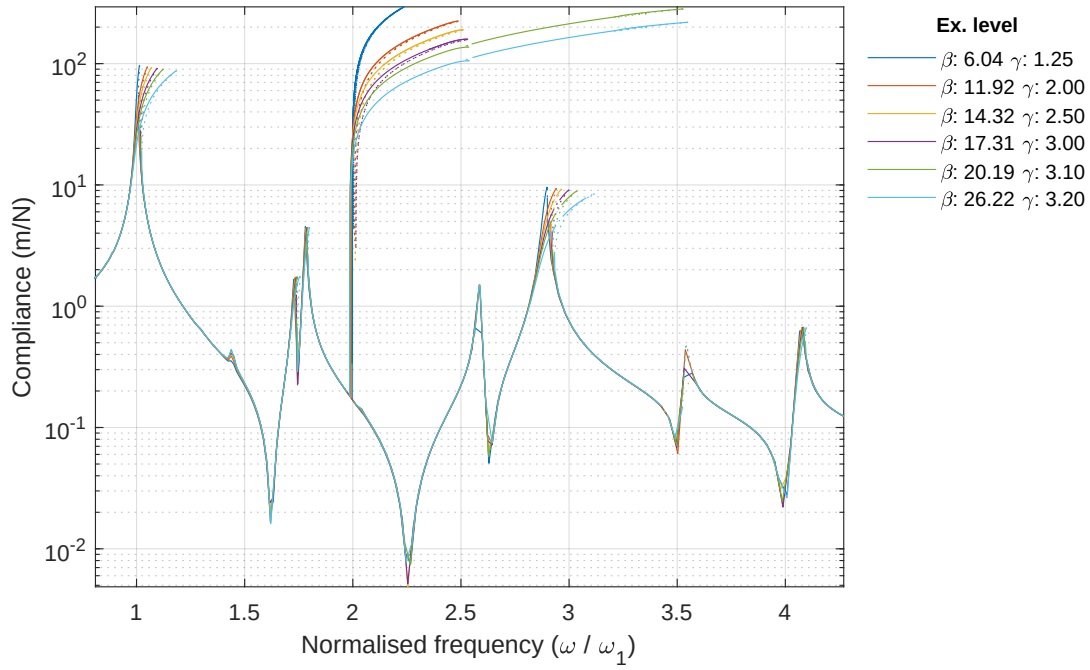


(b) Simulated frequency response of a membrane due to direct and parametric excitation. The model includes 11 modes

Figure 3-37: Frequency response of a clamped membrane showing locking and no shooting yet.

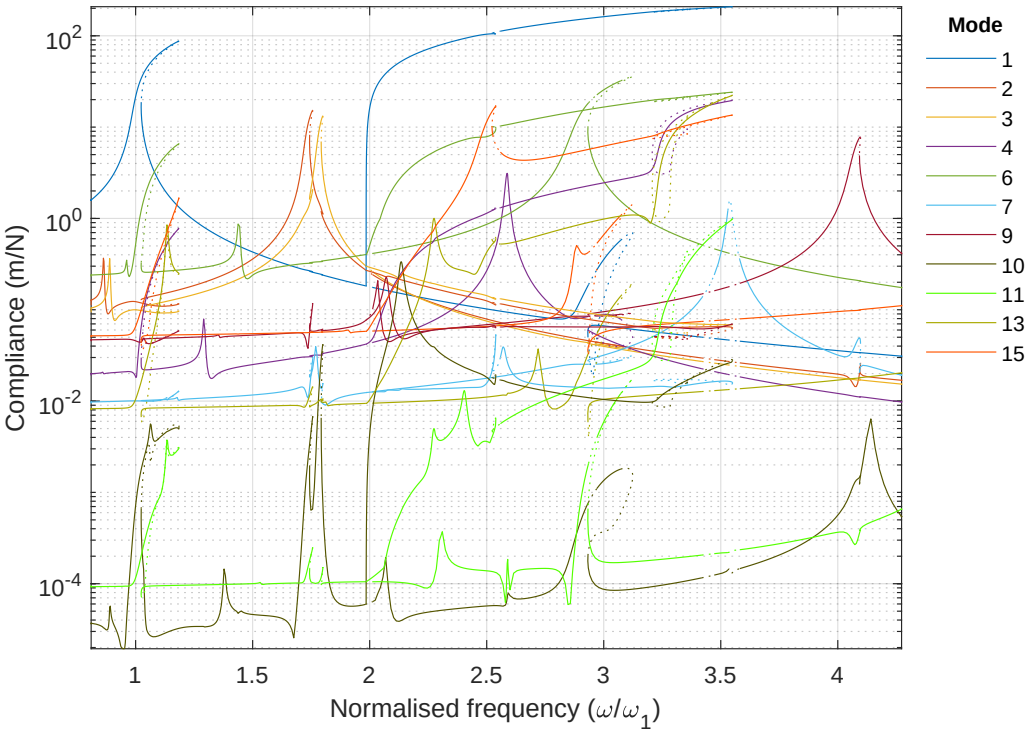


(a) Experimentally obtained frequency response for several forcing (voltage) levels. Note locking at $0.34\,\mathrm{V}$

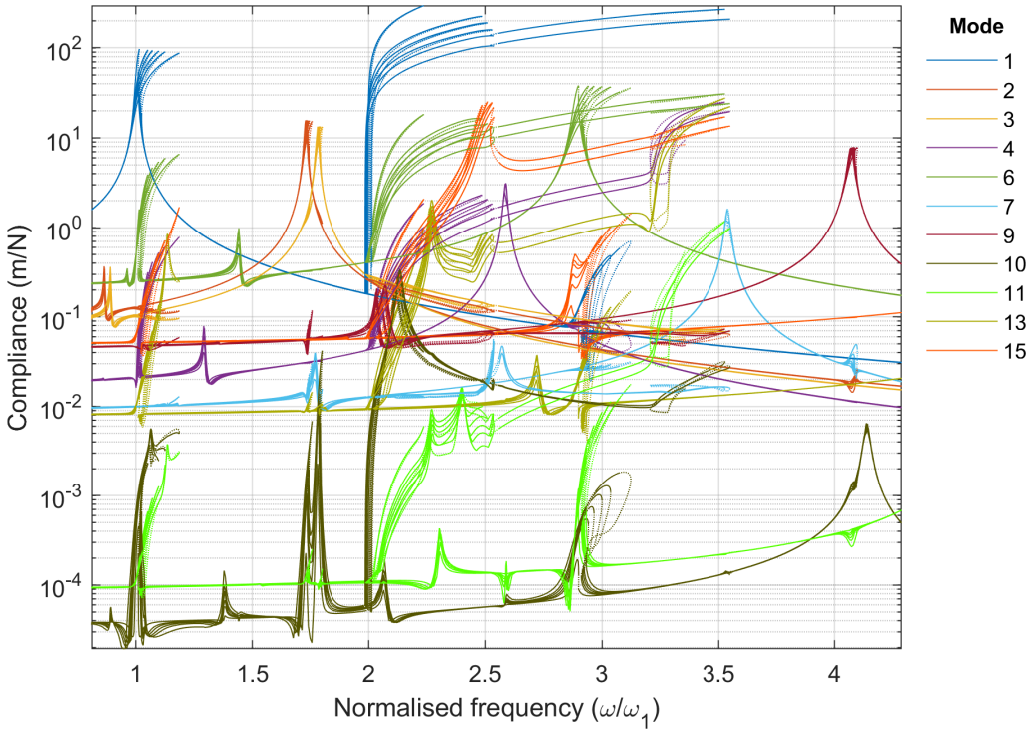


(b) Simulated physical frequency response for increasing direct (F) and parametric (γ) excitation. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-11}$,$\gamma_0 = 0.01$.

Figure 3-38: Frequency response of a clamped membrane showing shooting. The model includes 11 modes.



(a) Simulated modal frequency response for excitation level $\beta = 26.22$ and $\gamma = 3.2$.



(b) Simulated modal frequency response for all forcing levels.

Figure 3-39: Frequency response of a clamped membrane showing shooting.

Note that after the locking point (at $\omega = 2.53$), the period doubling branch is unstable for a short interval. This is also visible in the experimental data, as the amplitude fluctuates at this point. To investigate this behaviour, all found bifurcations are shown in Figure 3-40. After the locking point, a torus bifurcation shows up, enclosing the unstable interval.

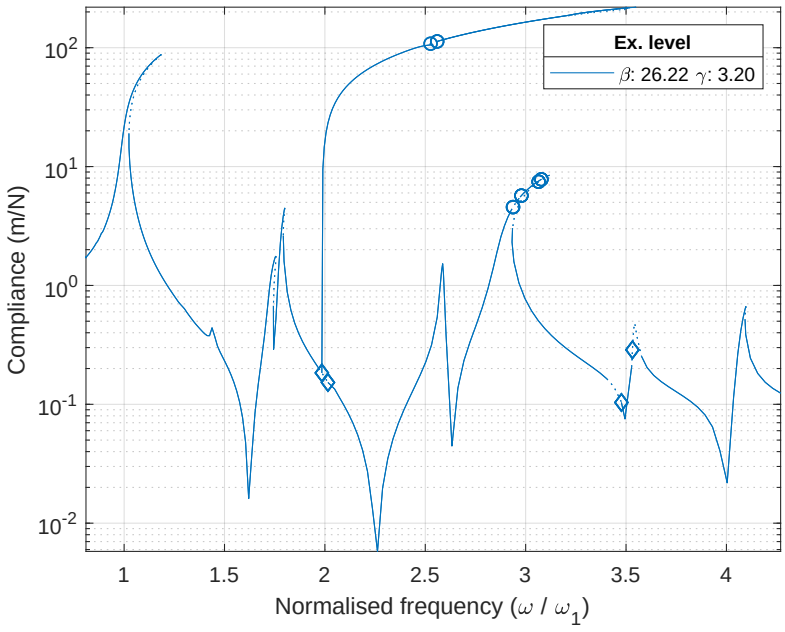
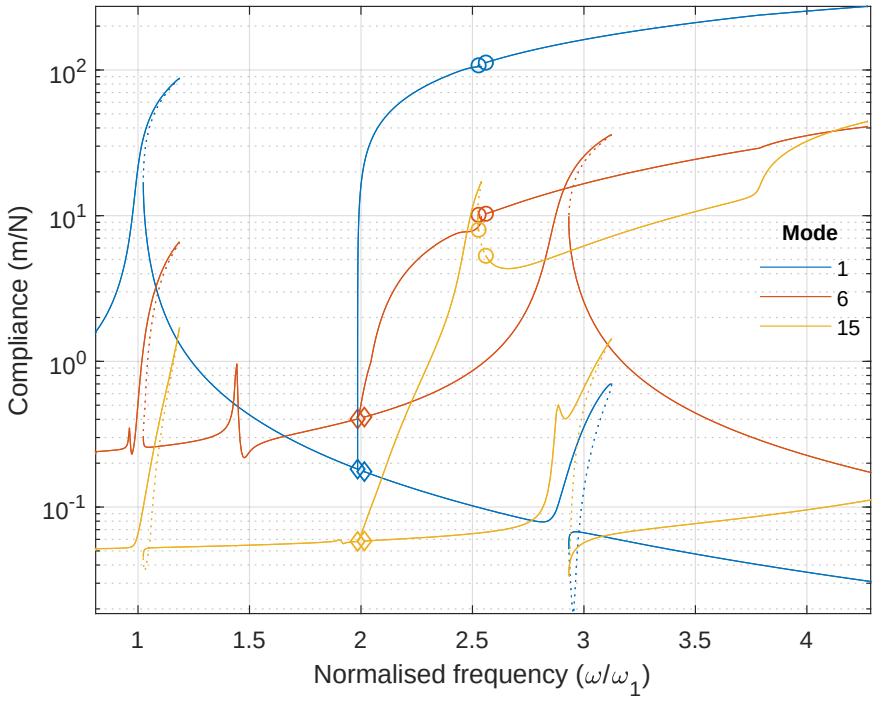
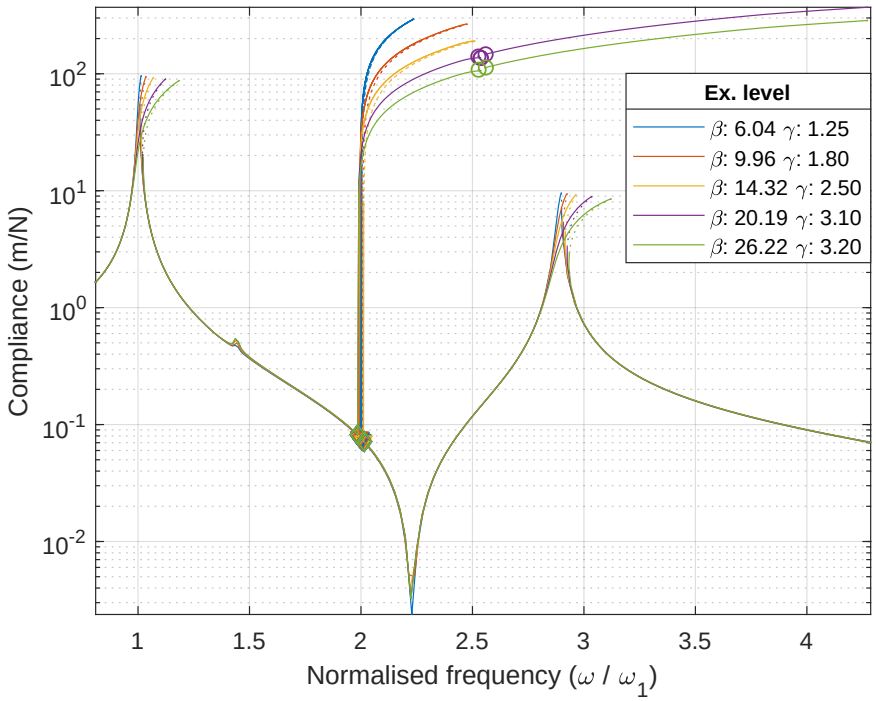


Figure 3-40: Simulated frequency response incl. bifurcation points. Diamond: Period doubling bifurcation - Circle: Torus bifurcation. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-11}$,$\gamma_0 = 0.01$.

Note that, for the locking at $\omega = 2.53$, although only mode 15 shows a real resonance peak at high amplitude, also mode 6 is essential for the internal resonance (if only mode 15 is included in the ROM, no locking happens). If the ROM contains only the modes 1,6 and 15, the locking - shooting phenomena can be observed (see Figure 3-41). However, the shooting branch now continues without any obstacles after it has overcome the locking at $\omega = 2.53$, as the modes required for the soft locking at $\omega = 3.24$ are not present.



(a) Simulated modal frequency response for excitation level $\beta = 26.22$ and $\gamma = 3.2$.



(b) Simulated frequency response incl. bifurcation points. Diamond: Period doubling bifurcation - Circle: Torus bifurcation. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-11}$,$\gamma_0 = 0.01$.

Figure 3-41: Frequency response of a clamped membrane showing shooting. The model consist of mode 1,6 and 15.

In Table 3-34, the eigenfrequencies of the interacting modes at the locking points are shown. In appendix B-1-2, the ratio is compared to what is found from Fourier analysis of the solution at the locking points (Figure B-4 and Figure B-5). The time signals of the solution points are shown in Figure 3-43 and Figure 3-45. The resulting shape of vibration is shown in Figure B-6 and Figure B-7 in appendix B-1-2. A cross section of this shape along the whole period doubling resonance curve is shown in Figure 3-46.

Table 3-34: Internal resonance along the period doubling resonance of the first mode. The actual frequency of vibration of a mode is denoted as ω^*, and the eigenfrequencies ω_i correspond to those listed in Table 3-33. Note that, due to parametric excitation, the frequency at which the fundamental mode vibrates is half the parametric excitation frequency ($\omega_1^* = \frac{1}{2}\omega_f$).

ω_f/ω_1	Interacting modes	Eigenfrequency ratio	Frequency ratio from FFT
2.53	1,6,15	$\omega_6 = 2.28\omega_1^*, \omega_{15} = 4.58\omega_1^*$	$\omega_6^* = 3\omega_1^*, \omega_{15}^* = 5\omega_1^*$
3.24	1,4,13	$\omega_4 = 1.6\omega_1^*, \omega_{13} = 3.36\omega_1^*$	$\omega_4^* = 2\omega_1^*, \omega_{13}^* = 4\omega_1^*$

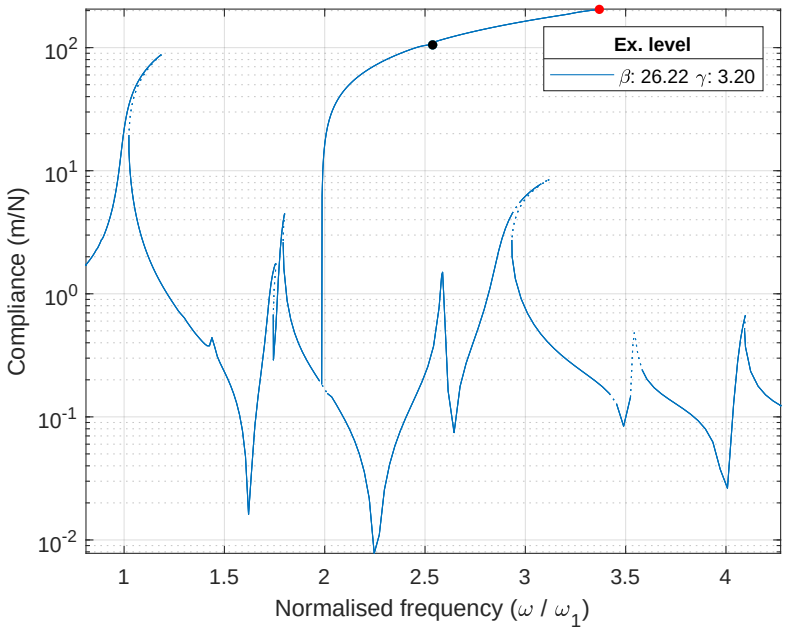


Figure 3-42: Simulated frequency response for increasing direct (F) and parametric (γ) excitation. For the black and red dot, the solution is further investigated. Excitation levels: $F_i = F_0\beta$ and $\gamma_i = \gamma_0\gamma$, where $F_0 = 1 \times 10^{-11}$,$\gamma_0 = 0.01$.

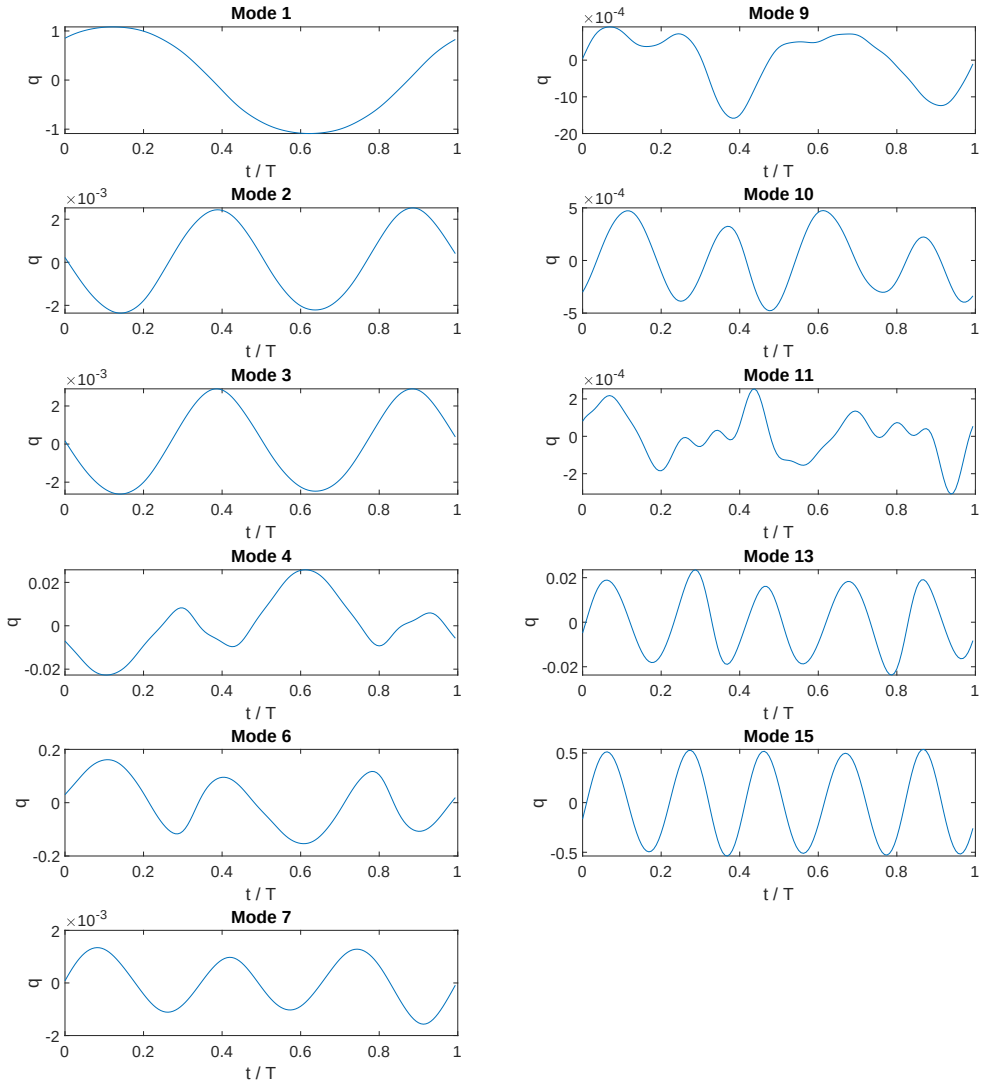
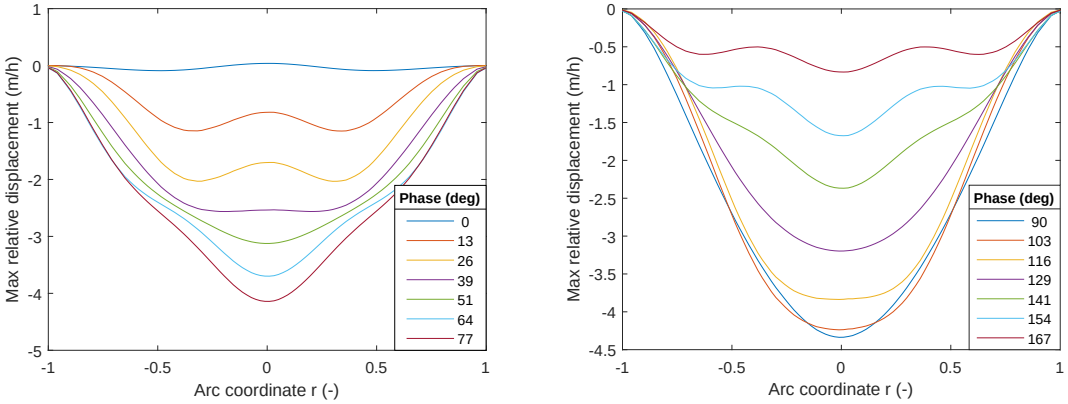
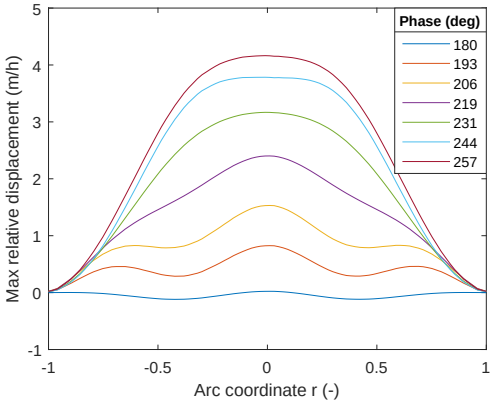


Figure 3-43: Modal time signal of stable solution along the resonance curve at $\omega_f = 2.53\omega_1$ (the red dot in Figure 3-42)

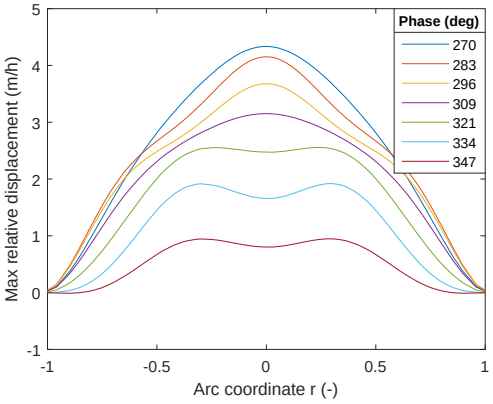


(a) 1st quarter period

(b) 2nd quarter period



(c) 3rd quarter period



(d) 4th quarter period

Figure 3-44: Deformation along cross section during one period at $\omega_f = 2.53\omega_1$

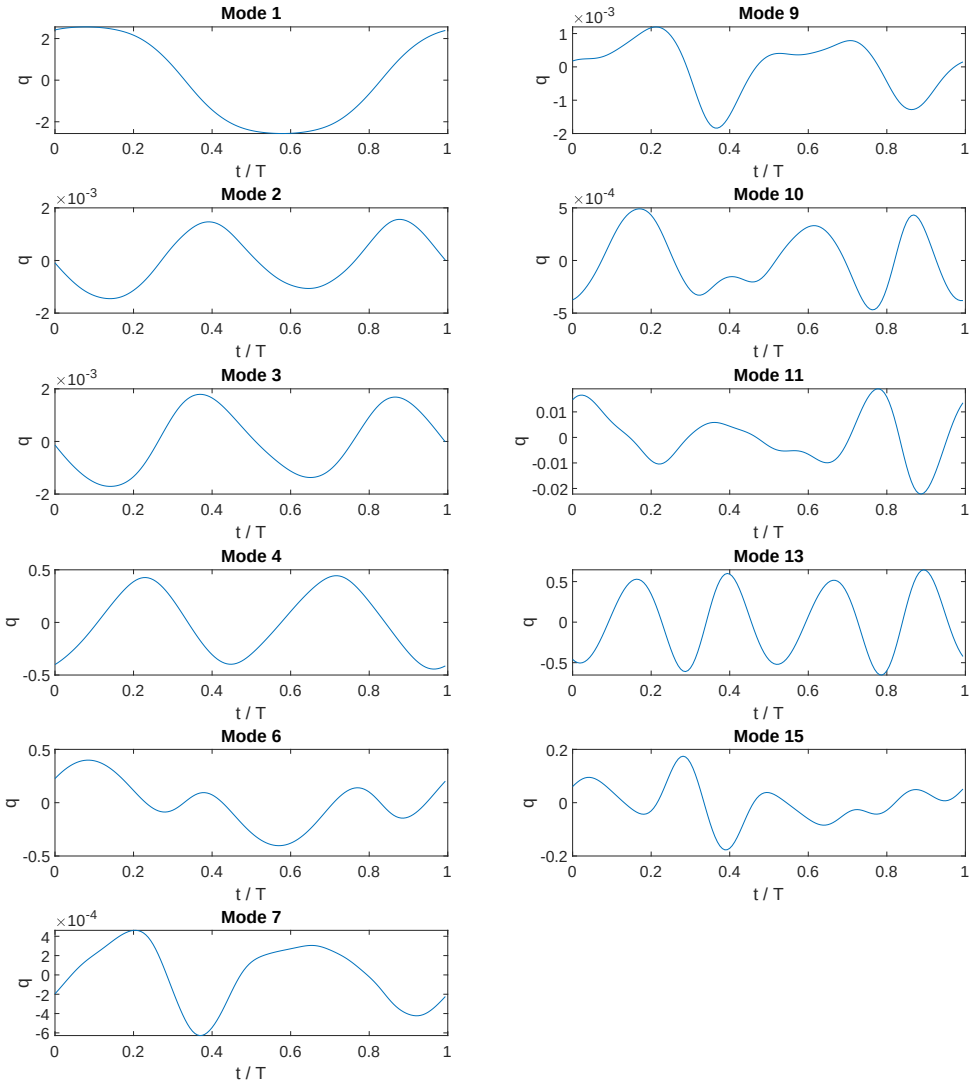


Figure 3-45: Modal time signal of stable solution along the resonance curve at $\omega_f = 3.37\omega_1$ (the black dot in Figure 3-42)

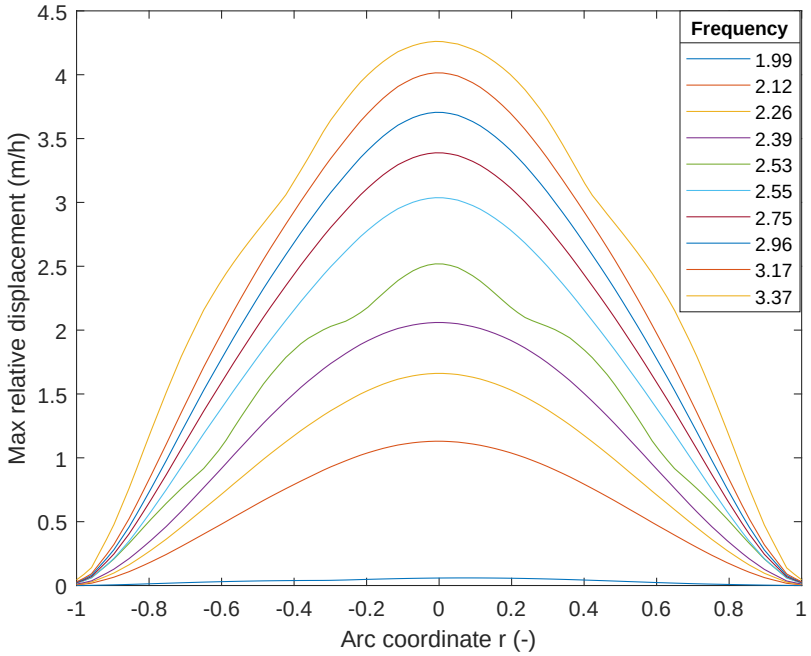


Figure 3-46: Vibration amplitude along cross section $x = 0$ for different points along the parametric resonance peak.

Conclusion

Regarding the preceding analysis, it can be concluded that the nonlinear ROM yielding from the condensed STEP method is able to describe very complicated experimentally observed effects like shooting. Those effects appear directly, without tweaking the modal equations of motion.

Beside, it may be observed that the parametric resonance remains bounded due to modal interaction, causing additional effective damping (In literature [18], usually artificial nonlinear damping like $F_{\mathrm{d}} = x^2\dot{x}$ is introduced). Therefore, mode-coupling damping can be seen as an essential mechanism to describe parametric resonance correctly.

3-4 FPU

The famous FPU problem [21] states that for a set of nonlinear coupled oscillators, energy is not equally divided after infinite time. Moreover, the system tends towards the initial conditions U_0 after a time T_1. Every $T_2 = nT_1$, the initial conditions are met almost exactly. The original problem was posted for a string where the individual atoms where modelled using linear and nonlinear reaction forces. Because the real (nonlinear) forces between the atoms were unknown, a qualitative model consisting of quadratic, cubic or discontinuous forces was introduced.

3-4-1 FPU in strings

Using our code, the equations of motion for a string can be derived from first principle. Using an ODE solver, the behaviour monitored at a large time scale. In this way, the behaviour can be compared to the FPU observations.

$\tilde{E}$ is taken as measure for the nonlinearity of the initial state. $\tilde{E}$ represents the energy (both kinetic and potential) in the initial state per mode, relative to the energy needed to pre-stress the membrane:

$$\tilde{E} = \frac{E_{\text{tot}}}{n_{\text{mode}} E_{\text{pre}}} \tag{3-13}$$

Here, E_{pre} equals the energy needed to pre-stress the membrane and E_{tot} equals the total energy in the system (of all modes). Another, more absolute measure of the initial state is known as the equivalent noise temperature. From a thermodynamic point of view, the energy is seen as a source of noise, related to the entropy of the system. The equivalent noise temperature writes (where k_{B} equals the Boltzmann constant, see [4]):

$$T_{\text{noise}} = \frac{E_{\text{tot}}}{n_{\text{mode}} k_{\text{B}}}; \tag{3-14}$$

The initial conditions are chosen such that all modes have an equal amount of energy. All modes start in their equilibrium position (zero displacement), having a velocity corresponding to their initial energy level.

In Figure 3-47, Figure 3-49 and Figure 3-50, the time response is shown for different initial conditions (structure properties are listed in Table 3-35, simulation settings are listed in Table 3-36). For low initial energy, there is only weak interaction. In Figure 3-48, a slow oscillation is visible, which grows in time and probably would tend towards chaos. For high initial energy, this chaotic behaviour appears directly (Figure 3-50), and energy flows freely from one mode to another. As stated in the FPU problem, there does not appear a steady state, in which the system would rest until infinite time.

Table 3-35: Structure parameters

Parameter	Value	Description
E	210 GPa	Youngs modulus
σ	1520 MPa	Pre-stress
D	0.001 m	Diameter
L	1 m	Length
ρ	7850 kg m^{-3}	Density

Table 3-36: Simulation settings for long-time behaviour of a string. Note that χ is chosen such that the each mode has the same initial energy.

Parameter	Value	Description
n_{modes}	6	Number of modes in the reduced order model
ζ	5e-10	Viscous damping active on all modes
n_{per}	500	number of periods simulated
χ	1/6	Fraction of initial energy in fundamental mode

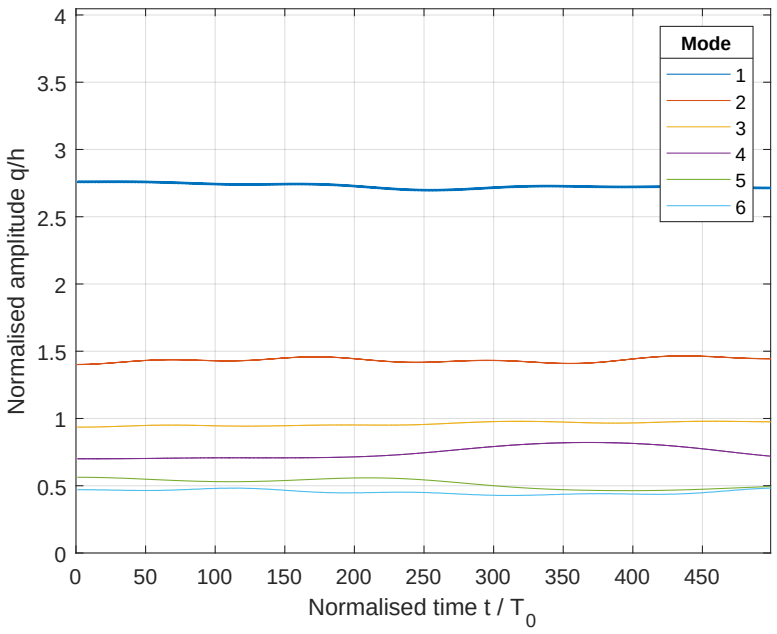


Figure 3-47: Long-time behaviour of a string: RMS-amplitude of vibration. $\tilde{E} = 0.016$

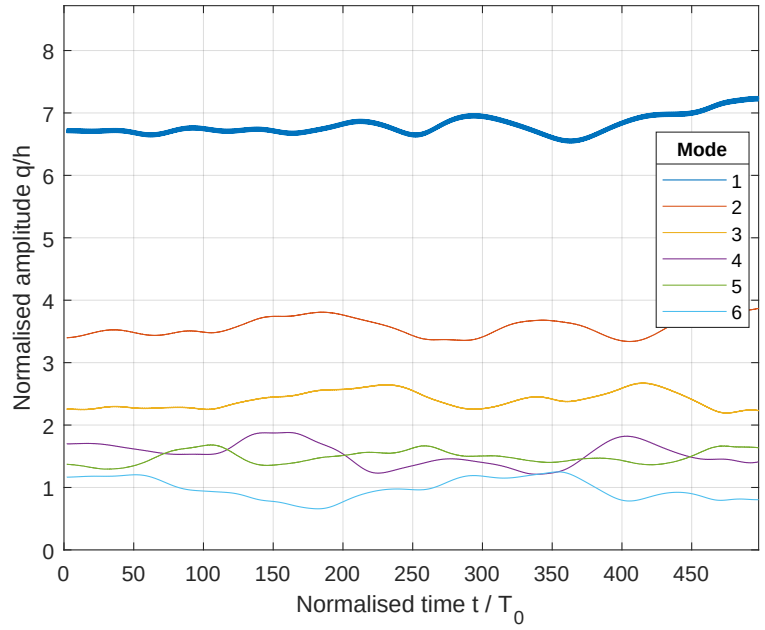


Figure 3-48: Long-time behaviour of a string: RMS-amplitude of vibration. $\tilde{E} = 0.10$

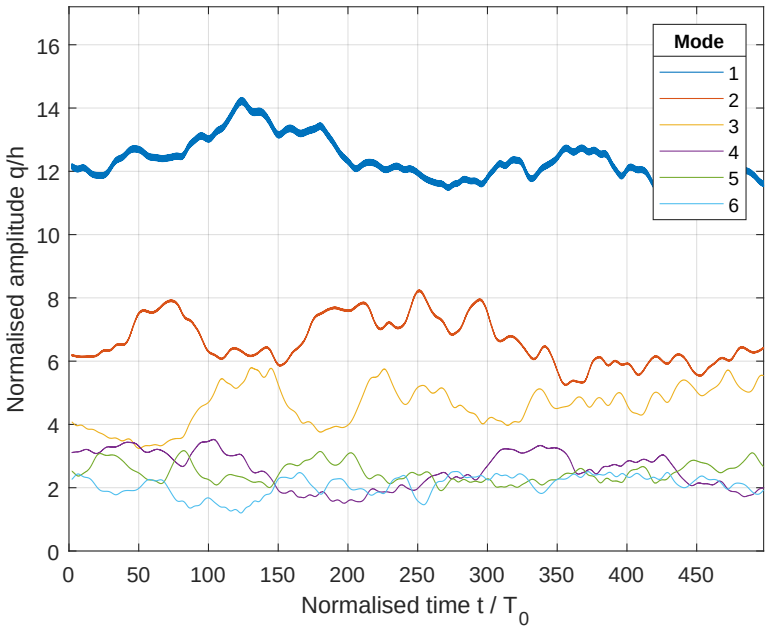


Figure 3-49: Long-time behaviour of a string: RMS-amplitude of vibration. $\tilde{E} = 0.41$

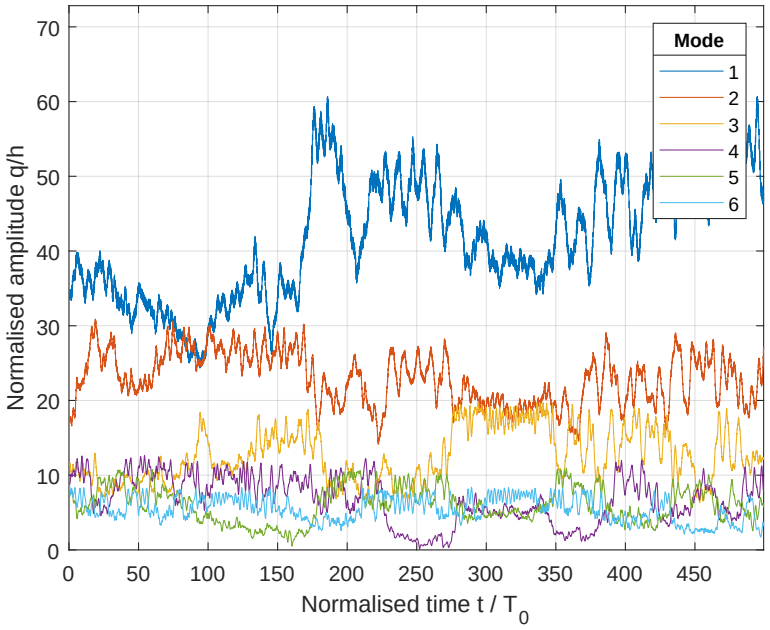


Figure 3-50: Long-time behaviour of a string: RMS-amplitude of vibration. $\tilde{E} = 10.2$

3-4-2 FPU in membranes

For a membrane, the same simulations are performed. The results are shown in Figure 3-51, Figure 3-52 and Figure 3-53, simulation setting are listed in Table 3-38 and structure parameters are given in Table 3-37. Interestingly, again slow oscillations appear, already at low energy levels (Figure 3-51). In contrary to the string, now those oscillations remain stable, also for larger energy levels (Figure 3-52). Finally, when increasing the energy enough, the system becomes chaotic (Figure 3-53), and the slow periodicity in amplitude is lost.

Table 3-37: Structure parameters

Parameter	Value	Description
E	1044 GPa	Youngs modulus monolayer graphene
ν	0.16	Poisson's ratio
T_0	$0.035\,\mathrm{N\,m^{-1}}$	Pretension
h	0.335 nm	Membrane thickness
R	30 nm	Radius

Table 3-38: Simulation settings for long-time behaviour of a membrane.

Parameter	Value	Description
n_{mode}	6	Number of modes in the reduced order model
ζ	5e-10	Viscous damping active on all modes
n_{per}	500	number of periods simulated
χ	1/6	Fraction of initial energy in fundamental mode

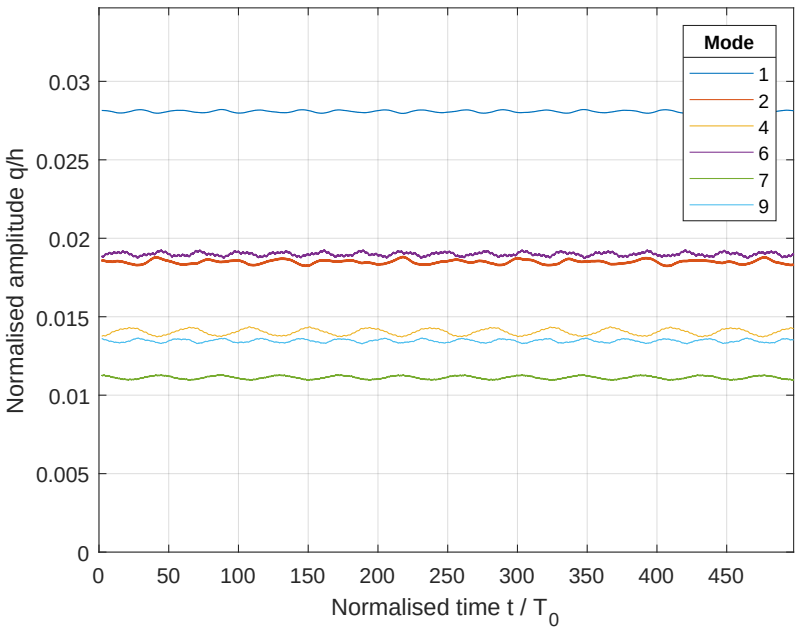


Figure 3-51: Long-time behaviour of a membrane: RMS-amplitude of vibration. $\tilde{E} = 0.0042$ or $T = 1K$

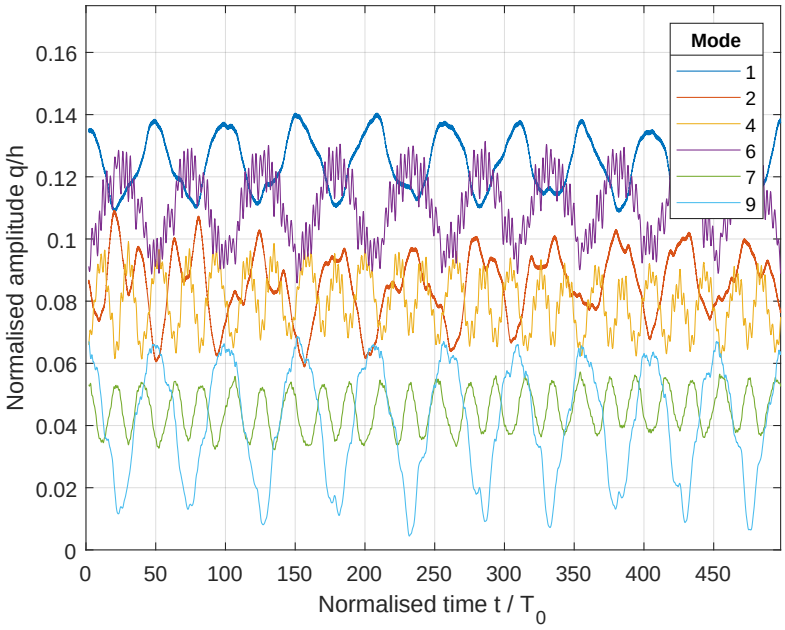


Figure 3-52: Long-time behaviour of a membrane: RMS-amplitude of vibration. $\tilde{E} = 0.084$ or $T = 20K$

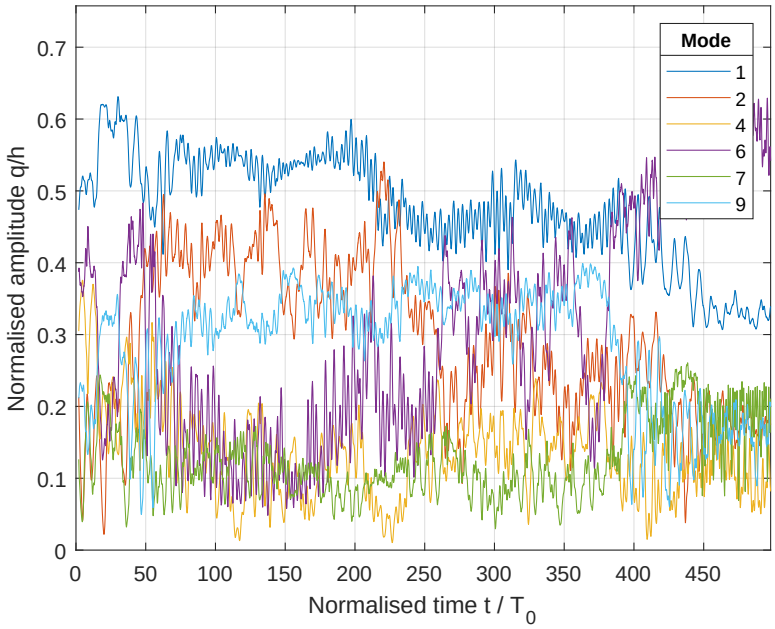


Figure 3-53: Long-time behaviour of a membrane: RMS-amplitude of vibration. $\tilde{E} = 1.67$ or $T = 400K$

Coefficient of variance

From Figure 3-51 and Figure 3-52, it may be observed that for larger initial velocity, the fluctuation in amplitude of all modes increase. The fluctuation in amplitude can be described by the coefficient of variance (or relative standard deviation), which is defined as the standard deviation divided by the mean:

$$c_{\mathrm{v}} = \frac{\sigma}{\mu} \tag{3-15}$$

This coefficient is calculated for the energy of the fundamental mode of each simulation. For different initial conditions, the result is shown in Figure 3-55 for a membrane, and in Figure 3-54 for a string. For the membrane, the coefficient of variance is also plotted versus the equivalent noise temperature in appendix B-1-3.

From Figure 3-55, it becomes clear that the slow oscillation amplitude increases linearly with energy. However, when the relative energy per mode ($\tilde{E}$) approaches unity, the system turns into chaos, and the coefficient of variance saturates near unity. Note that the string (Figure 3-54) shows the same trend, having only a slight offset.

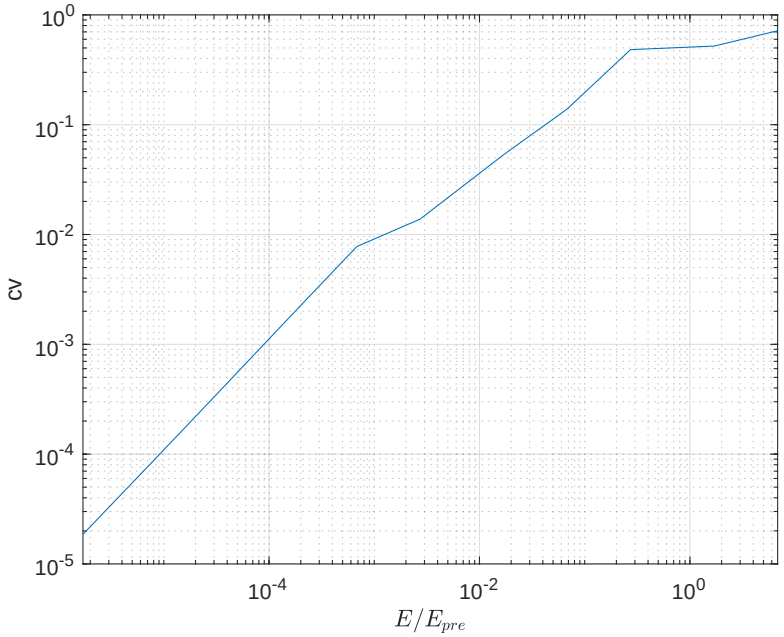


Figure 3-54: Coefficient of variance as a function of initial energy for a string.

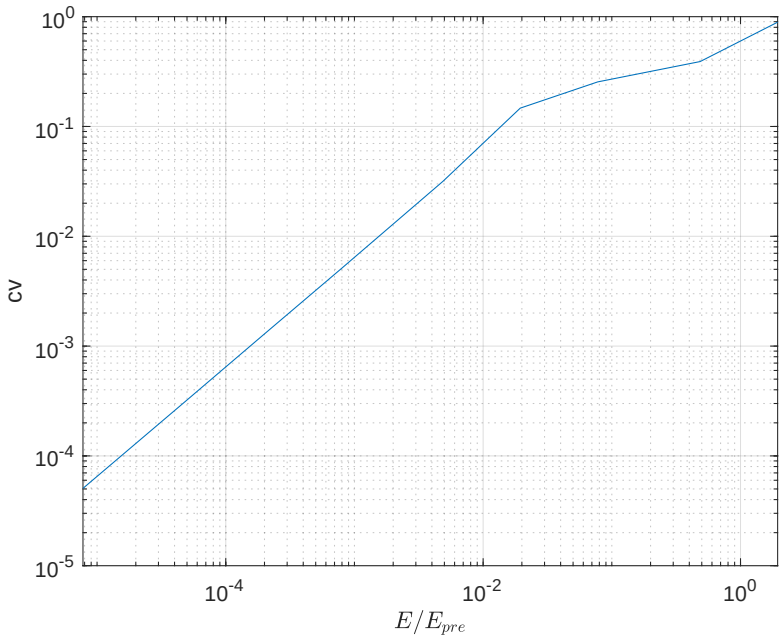


Figure 3-55: Coefficient of variance as a function of initial energy for a membrane.

Conclusion

Regarding the original FPU problem, some features where found in the analysed structures. As the FPU problem stated, there is, (at sufficient high energy level), no steady state configuration in which the system rests after very long time (thermalization does not happen). Moreover, the membrane shows a periodicity of it's amplitude of vibration on a large timescale. Here, the initial conditions are met closely every period (the system remembers it's initial state). This last feature is not so clear observed for a string.

3-5 Overview

In Table 3-39, an overview is provided of all observed dynamic effects investigated in this chapter. For some structures, a certain effect is not investigated, so it is left open. The effect of interacting modes increasing very rapidly at a certain frequency as well as shooting are named under internal resonance, as this is the expected mechanism behind. For a string including in-plane modes, internal resonance is described in the appendix, section D-6-2.

Table 3-39: Overview of the observed nonlinear dynamic effects for different structures. Unknown combinations are left open. Structures marked with a star (*) have additional bending stiffness. †: this effect can be found in the appendix, section D-6-2

Element	Softening	Hardening	Internal resonance	FPU
String	×	✓		×
String (incl. in-plane modes)	✓	✓	✓†	
Circular membrane	×	✓		✓
Circular membrane*	×	✓	✓	
Square membrane	×	✓		
Square membrane*	×	✓	✓	
Cantilever	×	✓		

Chapter 4

Quality factor

This chapter focusses on the quality factor due to mode interaction. Usually, the quality factor is determined by physical damping mechanisms, which convert vibration energy into heat. However, in coupled systems, energy can also flow from one mode to another, resulting in a decay of the drained mode comparable to physical damping mechanisms. This effect can be observed during the transient response of a freely vibrating system, which is therefore the main focus in this section.

In section 4-1, the system of equations is studied, in order to understand the (mathematical) mechanisms behind the mode interaction. Those equations can hardly be solved analytically, but the dependence on different parameters can be estimated. The influence of initial conditions (total and noise energy) and shape (radius and aspect ratio) is investigated in the succeeding sections.

4-1 Free vibration

In linear systems, all eigenmodes are uncoupled, so energy cannot flow from one mode to another. In coupled (nonlinear) systems however, energy can flow between the modes. The rate of flow cannot be deduced from the equations directly. Therefore, in this section estimations are made for the scaling of the power.

4-1-1 Equations

Consider a ROM consisting of 2 modes, having only cubic nonlinear stiffnesses (like the straight string in section 3-1-2, or the flat membrane from section 3-1-4) and no external forcing. Then, from Eq. (3-1), only the following remains:

$$\begin{aligned} m_1\ddot{q}_1 + k_1 q_1 &= -b^{(1)}_{111}q_1^3 - b^{(1)}_{112}q_1^2 q_2 - b^{(1)}_{122}q_1 q_2^2 - b^{(1)}_{222}q_2^3 \\ m_2\ddot{q}_2 + k_2 q_2 &= -b^{(2)}_{111}q_1^3 - b^{(2)}_{112}q_1^2 q_2 - b^{(2)}_{122}q_1 q_2^2 - b^{(2)}_{222}q_2^3 \end{aligned} \tag{4-1}$$

Note that there are no linear damping terms present in Eq. (4-1), as we are only interested in damping due to nonlinear effects. Therefore, in this chapter the total energy is conserved.

The first term on the right-hand-side in the first line of Eq. (4-1) ($b^{(1)}_{111}q_1^3$) acts as a cubic stiffness solely on the first mode (q_1), and therefore creates no coupling to the second mode. The last 3 terms on the right-hand-side (with stiffness $b^{(1)}_{112}, b^{(1)}_{122}, b^{(1)}_{222}$) have mixed indices, and therefore act as a nonlinear coupling spring. Through this coupling, energy can flow from one mode to another. The rate of energy flow (power) can, at first glance, not easily be obtained from the equations. To get some insight, the elastic potential is derived in the next section, which appears to be very useful later on.

Elastic potential

From the stiffness terms in Eq. (4-1), an elastic potential can be derived. Adding the kinetic energy, the total energy in the system can be determined. The linear elastic energy in mode 1 writes:

$$E^{(1)}_{\mathrm{L}} = \frac{1}{2}k_1 q_1^2 \tag{4-2}$$

The nonlinear elastic potential of each spring can be found by integrating the force (further elaborated in section C-2-1):

$$E_{1112} = \int F\mathrm{d}u = \int b^{(1)}_{112}q_1^2 q_2 \mathrm{d}q_1 = \frac{1}{3}b^{(1)}_{112}q_1^3 q_2 \tag{4-3}$$

The total nonlinear potential (of the whole system) writes:

$$E_{\mathrm{NL}} = \frac{1}{4}b^{(1)}_{111}q_1^4 + \frac{1}{3}b^{(1)}_{112}q_1^3 q_2 + \frac{1}{2}b^{(1)}_{122}q_1^2 q_2^2 + b^{(1)}_{222}q_1 q_2^3 + \frac{1}{4}b^{(2)}_{222}q_2^4 \tag{4-4}$$

Here, the energies with mixed indices allow for coupling (E_{NLC}), the others only for storage (E_{NLS}).

$$E^{(1)}_{\mathrm{NLS}} = \frac{1}{4}b^{(1)}_{111}q_1^4, \quad E^{(2)}_{\mathrm{NLS}} = \frac{1}{4}b^{(2)}_{222}q_2^4 \tag{4-5}$$

$$E^{1-2}_{\mathrm{NLC}} = \frac{1}{3}b^{(1)}_{112}q_1^3 q_2 + \frac{1}{2}b^{(1)}_{122}q_1^2 q_2^2 + b^{(1)}_{222}q_1 q_2^3 \tag{4-6}$$

Note that, because multiple stiffness terms yield from the same term in the elastic potential, those terms are related to each other (relations between terms can be found in [7]).

4-1-2 Energy transfer between modes

The coupling terms in the right hand side of Eq. (4-1) can be seen as nonlinear springs, connecting both modes. Each mode can add or subtract energy to the spring, allowing energy to flow from one mode to another. If we assume harmonic motion (with amplitude $\hat{q}$), the work done on a nonlinear spring can be estimated by integrating the power with respect to time (a derivation can be found in section C-2-3):

$$W = \int_0^T P \mathrm{d}t \tag{4-7}$$

The work of mode 1 on the cubic spring with coefficient $b_{112}^{(1)}$ (during one period of mode 1) can be determined now. Here, the frequency of mode 2 is expressed as a fraction of mode 1 $\omega_2 = \frac{n}{m}\omega_1$, yielding:

$$W_{112}^{(1)} = b_{112}^{(1)} \hat{q}_1^3 \hat{q}_2 \frac{1}{8} \left(\frac{\cos(2\pi m\delta)}{\delta} - \frac{\cos(2\pi m\delta_3)}{\delta_3} - \frac{\cos(2\pi m\sigma)}{\sigma} + \frac{\cos(2\pi m\sigma_3)}{\sigma_3} \right) \tag{4-8}$$

$$\delta = \frac{\omega_2 - \omega_1}{\omega_1},\ \delta_3 = \frac{\omega_2 - 3\omega_1}{\omega_1}, \sigma = \frac{\omega_1 + \omega_2}{\omega_1}, \sigma_3 = \frac{3\omega_1 + \omega_2}{\omega_1}\ ,\ T_1 = m\frac{2\pi}{\omega_1} \tag{4-9}$$

The expression can be split up in an energy and frequency depended part, multiplied with some constatnt c:

$$W_{112}^{(1)} = cE_{1112} f(\omega_1, \omega_2) \tag{4-10}$$

This is an important result, as it allows us to distinguish between the mechanisms creating the energy flow. The energy part is used to determine the influence of energy-related quantities like noise-temperature (section 4-5). The frequency dependent part plays a role when analysing shape influence (section 4-7).

Entropy: energy flow direction

In literature [22], a law is derived from thermodynamics, indicating the (average) energy flow between coupled mechanical oscillators:

$$\dot{W} = < E_1 > - < E_2 > \tag{4-11}$$

where $< E_i >$ indicates the average modal energy. This law ensures that, on time average, energy flows from a mode containing a lot of energy to one containing less, revealing the macroscopic driving force behind the damping mechanism.

4-2 Simulation approach

In this section, the simulation approach is outlined. First, a proper measure for damping is discussed, followed by the used simulation setup. Finally, an example is given to illustrate the procedure. Beside, in section C-1-1, the numerical accuracy of the used ODE solver is discussed.

Damping measure

Damping can be expressed in several measures, but one close to physics is the loss factor:

$$\eta = \frac{\Delta U}{2\pi U} \tag{4-12}$$

where U equals stored energy (in the mode under study) and ΔU the lost energy (from the mode under study towards the other modes) during one period of the mode. Note that η is usually used for linear systems. In that case it is independent of amplitude, in nonlinear systems this can change.

For some physical sources of damping, η can be determined from first principles (like thermoelastic damping). However, the energy flow between modes cannot easily be deduced from the governing equations. Therefore, it has to be obtained empirical (numerical or experimentally). In this work, only numerical simulations are performed, which are compared to experiments from literature if available.

Simulation setup

The simulations comprise the following:

1. Initial conditions are determined from global settings (like energy levels etc.)
2. The response of the system (in absence of external forces) is calculated for a certain timespan.
3. The effective damping is determined from the response.

This is close to what is called a Ringdown experiment in literature. The simulated time interval is quite short, such that the system remains close to the global properties set by the initial conditions. In this way, the Q-factor is effectively evaluated at one amplitude and the amplitude dependence of Q in the nonlinear system can be evaluated. To ensure this, the number of simulated periods should remain shorter than the quality factor ($n_{\text{per}} < Q$). In this way, the influence of the initial settings can be investigated.

From the initial settings, energy levels for all modes can be determined. Those can be translated to initial velocity $v(0)$ and displacement $u(0)$ by specifying the initial phase θ (assuming all modes to vibrate at their own eigenfrequency). Assuming sinusoidal motion, the initial phase can be determined:

$$\begin{aligned} u &= \sin(\omega t + \theta) \\ v &= \omega \cos(\omega t + \theta) \end{aligned} \tag{4-13}$$

$$\theta = \arctan\left(\frac{u(0)}{v(0)/\omega}\right) \tag{4-14}$$

However, this initial phase can influence the behaviour significantly. To remove this influence, each simulation is repeated (several times) using different initial phase (obtained from a random set) for each mode each simulation. From each batch of datasets belonging to the same initial settings, statistical measures like the mean and variance can be determined, finally providing use the desired result. The number of simulations per parameter set (n_r) is shown for each analysis, and varies between 40 and 200, depending on the accuracy needed.

To model the transient response, an ODE solver is used, which integrates the system of equations over a given time interval. The implementation of the solver is described in section 2-5-4.

Extracting the loss-factor η

The loss-factor can be calculated at every time-step (further explained in appendix section C-2-4):

$$\eta = \frac{\Delta U}{2\pi U} = \frac{PT}{2\pi U} = \frac{P}{\omega U} \tag{4-15}$$

Because the system is nonlinear, the vibration frequency changes with amplitude. Therefore, instead of just assuming the linear eigenfrequency, the instantaneous frequency (see section C-2-5) is determined, which can be seen as the geometric mean of the Amplitude Spectral Density (ASD):

$$f_\mathrm{inst} = \frac{\int_0^\infty f X(f) \mathrm{d}f}{\int_0^\infty X(f) \mathrm{d}f} \tag{4-16}$$

Here, $X(f)$ equals the Fourier transform of the time signal, and is determined using the Fast Fourier Transform (FFT) algorithm (this is done for multiple time intervals using a moving window).

Overview simulation setup

In Figure 4-1, an overview is presented of the simulation setup.

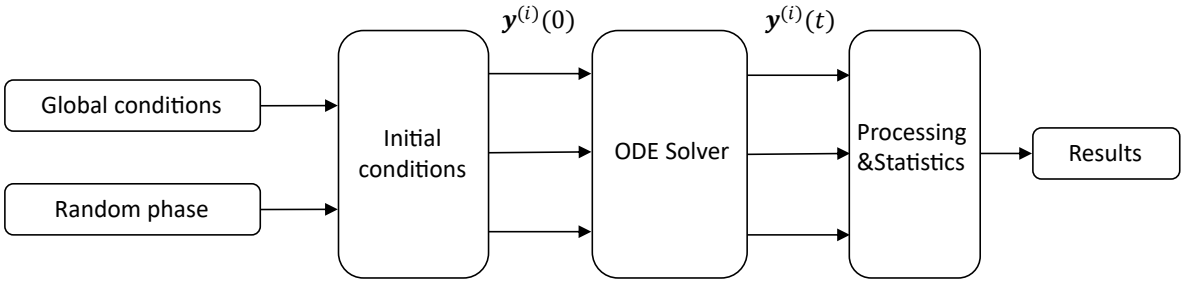


Figure 4-1: Overview of the simulation setup. Initial state vectors are denoted as $y^{(i)}(0)$, solution vectors as $y^{(i)}(t)$. Calculation of both loss-factor η and instantaneous frequency f_{inst} are performed during the processing.

4-2-1 Example

To illustrate the procedure, one simulation is shown in Figure 4-2. The simulation settings are listed in Table 4-1.

Note that there is very little viscous damping ($\zeta = 5 \times 10^{-10}$) added for numerical consistency. It is preferred here to have a very small damping source which is known and can be monitored, instead of 'unpredictable' numerical damping (due to ODE tolerances or machine precision), which can fluctuate for different simulations. Numerical damping is always present, but at least it is not dominant in this case. From ζ, the range in which results are valid can be estimated. This is further elaborated in section C-1-1.

Here, some key values are introduced which define the simulation settings. The fraction of the total initial energy present in the fundamental mode is defined as χ:

$$\chi = \frac{E_1}{E_{\text{tot}}} \tag{4-17}$$

The remaining energy is equally distributed over the other modes. As a measure for the initial energy, $\tilde{V}_0$ is defined as the relative initial velocity of the fundamental mode. The value corresponds to the amplitude of vibration for sinusoidal motion (like Eq. (4-13)), normalised to the structure thickness h:

$$\tilde{V}_0 = \frac{v_1(0)}{h\omega_1} \tag{4-18}$$

Using this definition, a value of $\tilde{V}_0 \ll 1$ corresponds to the linear regime, and $\tilde{V}_0 \gg 1$ to the nonlinear regime.

Table 4-1: Simulation settings method example.

Parameter	Value	Description
n_{modes}	20	Number of modes in the reduced order model
ζ	5×10^{-10}	Linear viscous damping for each mode (for numerical consistency)
n_{r}	5	Number of simulations per batch
n_{per}	200	Number of periods simulated
χ	0.6	Fraction of total initial energy in the fundamental mode
$\tilde{V}_0$	14	Relative initial velocity of the first mode

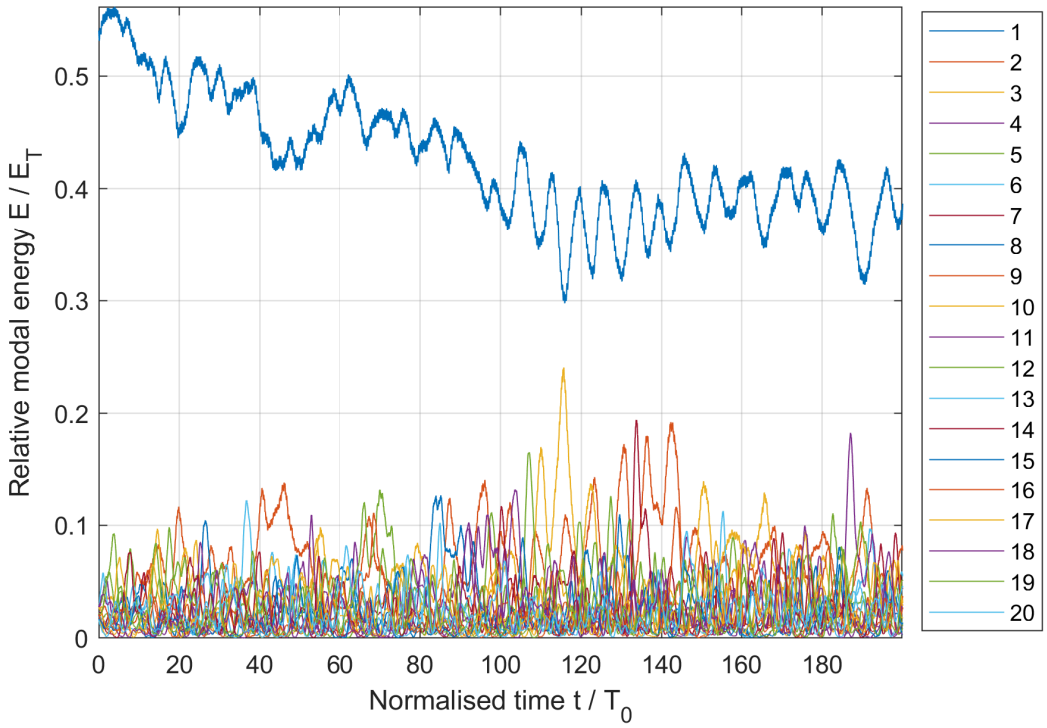


Figure 4-2: Relative modal energy: E equals the modal energy in each mode, E_T the total energy in the system, t equals time which is normalised to T_0, the period of the fundamental mode ($T_0 = 2\pi/\omega_1$). Different colours correspond to different modes.

4-2-2 Damping measures

Time-averaged loss factor

From the previous example, it becomes clear that there is no gradual decrease, it is very bumpy. Therefore, two descriptive numbers are defined, one corresponding to the time-averaged energy decay, and one to the instantaneous fluctuations. For one simulation, the loss factor η can be calculated at every timestep t_i. Calculating the time average of η yields the time-averaged loss factor η_{ta}:

$$\eta_{\text{ta}} = \frac{1}{N_{\text{s}}} \sum_{i=1}^{i=N_{\text{s}}} \eta(t_i) \tag{4-19}$$

where $\eta(t_i)$ equals the loss factor evaluated at sample time t_i and N_{s} equals the number of samples. Note that this is closely related to comparing the initial and final energy level. However, by ignoring the data in between, fluctuations in vibration frequency would not be taken into account.

Instantaneous loss factor

Calculating the standard deviation (σ) of η provides information of the dispersion. A constant decay should result in a small value, while a very irregular progress should yield a large standard deviation. The standard deviation is used to define the instantaneous loss factor:

$$\sigma = \sqrt{\frac{1}{N_{\text{s}} - 1} \sum_{i=1}^{N_{\text{s}}} \left(\eta(t_i) - \eta_{\text{ta}}\right)^2} \tag{4-20}$$

$$\eta_{\text{ins}} = \eta_{\text{ta}} + \sigma \tag{4-21}$$

From another point of view, η_{ins} can be seen as a soft upper bound for the loss factor, as approximately 84% of the set values should be lower. Note that η_{ins} is close related to the coefficient of variance c_{v} (defined in section 3-4-2), both being a nondimensional measure for the dispersion.

Batch data

As mentioned before, multiple simulations are run per batch, where each simulation has different random phase but the same initial energy levels. For all simulations in one batch, all data of η is taken together and regarded as one dataset, the batchdata. From this, the instantaneous and time-averaged loss factor are determined to provide a balanced statistical measure.

Note that for the standard deviation of multiple datasets, instead of Eq. (4-20), the pooled standard deviation would be right statistical measure. However, for data having large dispersion (small mean w.r.t. the sample data), the pooled version yields comparable results. As the loss factor is found to have large dispersion (Figure 4-2), the total standard deviation is assumed to still provide a good measure for the dispersion.

4-3 Membrane model

In the subsequent sections, damping through mode interaction is further investigated by performing simulations. The structure of interest is again a circular membrane made of graphene, as the damping in those structures is a significant obstacle for practical implementation [23]. Structure properties (as found from [23]) are listed in Table 4-2, eigenfrequencies are listed in Table 4-3 and Table 4-4.

Table 4-2: Properties circular graphene membrane

Parameter	Value	Description
E	$1014.9\,\mathrm{GPa}$	Youngs modulus graphene
ν	0.16	Poisson's ratio
T_0	$0.0340\,\mathrm{N\,m^{-1}}$	Pretension
h	$0.335\,\mathrm{nm}$	Membrane thickness
R	$300\,\mathrm{nm}$	Radius
ρ	$2209\,\mathrm{kg/m^3}$	Density

Table 4-3: Normalised eigenfrequencies of the first 9 axisymmetric modes of a circular membrane

Index	Waveindex(t,r)	$\omega_\mathrm{n}/\omega_1$
1	(0,1)	1
6	(0,2)	2.2959
15	(0,3)	3.6027
30	(0,4)	4.904
51	(0,5)	6.2114
74	(0,6)	7.5202
105	(0,7)	8.8341
140	(0,8)	10.152
175	(0,9)	11.479

Table 4-4: First 20 normalised eigenfrequencies of a circular membrane

Index	Waveindex(t,r)	$\omega_\mathrm{n}/\omega_1$	Shape
1	(0,1)	1	axisymmetric
2	(1,1)	1.5934	
3	(1,1)	1.5934	
4	(2,1)	2.1358	
5	(2,1)	2.1358	
6	(0,2)	2.2959	axisymmetric
7	(3,1)	2.6539	
8	(3,1)	2.6539	
9	(1,2)	2.9188	
10	(1,2)	2.9189	
11	(4,1)	3.1573	
12	(4,1)	3.1573	
13	(2,2)	3.5036	
14	(2,2)	3.5036	
15	(0,3)	3.6027	axisymmetric
16	(5,1)	3.6511	
17	(5,1)	3.6511	
18	(3,2)	4.0658	
19	(3,2)	4.0658	
20	(6,1)	4.1385	

4-4 Energy dependence

According to Eq. (4-10), the energy flow depends on the energy in the coupling springs. Some work in literature concerning this is shared in the next section (4-4-1). Thereafter, a theoretical hypothesis is constructed in section 4-4-2. To test it, simulations are performed subsequently in section 4-4-3.

4-4-1 Literature

In literature, the influence of energy in a system on mode coupling damping is already mentioned. Midtvedt et al. [4] reported a linear dependence of the loss factor on the total energy in the system. In Figure 4-3, their results are shown. Here, the noise energy is a constant portion of the total energy, and is translated to the equivalent noise temperature (reported on the x-axis). Despite their nice results, they do not provide a theoretical explanation for the observed behaviour. Therefore, this is investigated in the next section.

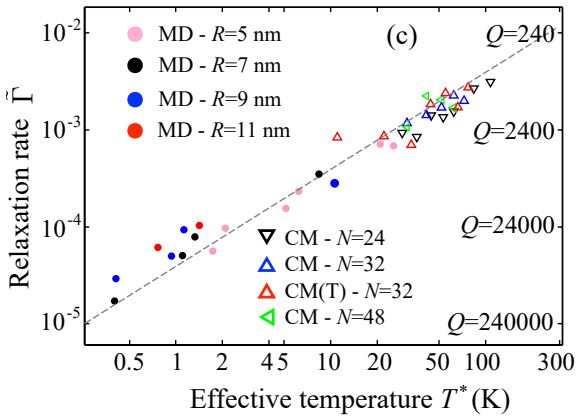


Figure 4-3: Energy dependence of the loss factor (of the fundamental mode). Results are obtained from Molecular Dynamics (MD) and Continuum Mechanics (CM). The noise energy is translated to the equivalent noise temperature. In the CM model, only axisymmetric modes are included.

4-4-2 Theory

According to Eq. (4-10), the energy flow depends on the energy in the coupling springs. This allows making a rough estimation of the loss factor due to this coupling. The loss factor is defined as:

$$\eta = \frac{\Delta U}{2\pi U}$$

The stored energy U equals the sum of both linear and nonlinear energy:

$$U = E_{\mathrm{L}} + E_{\mathrm{NLS}} \tag{4-22}$$

Recall Eq. (4-10) (the work done due to a coupling spring) to estimate the lost energy during one period:

$$\Delta U_{1112} = E_{1112} f_{1112}(\omega_1, \omega_2) \tag{4-23}$$

As our point of interest is currently the influence of energy levels, the frequency dependent term is ignored in the following part. Recall the nonlinear coupling and storage potentials:

$$E_{\mathrm{NLS}}^{(1)} = \frac{1}{4} b_{111}^{(1)} q_1^4, \quad E_{\mathrm{NLS}}^{(2)} = \frac{1}{4} b_{222}^{(2)} q_2^4 \tag{4-24}$$

$$E_{\mathrm{NLC}}^{1-2} = \frac{1}{3} b_{112}^{(1)} q_1^3 q_2 + \frac{1}{2} b_{122}^{(1)} q_1^2 q_2^2 + b_{222}^{(1)} q_1 q_2^3 \tag{4-25}$$

The energy in the coupling springs can be fixed to a constant fraction of the stored non-linear energy:

$$E_{\mathrm{NLC}}^{1-2} = \gamma E_{\mathrm{NLS}}^{(1)}, \tag{4-26}$$

yielding the energy loss in terms of stored nonlinear energy:

$$\Delta U \propto E_{1112} + E_{1122} + E_{1222} = E_{\mathrm{NLC}}^{1-2} = \gamma E_{\mathrm{NLS}}^{(1)} \tag{4-27}$$

Eq. (4-27) and Eq. (4-22) can be substituted in Eq. (4-12), to obtain the loss factor for the first mode:

$$\eta = \frac{\Delta U}{2\pi U^{(1)}} \propto \frac{\gamma E_{\mathrm{NLS}}^{(1)}}{E_{\mathrm{L}}^{(1)} + E_{\mathrm{NLS}}^{(1)}} = \gamma \frac{\frac{1}{4} b_{111}^{(1)} q_1^4}{\frac{1}{2} k^{(1)} q_1^2 + \frac{1}{4} b_{111}^{(1)} q_1^4} = \gamma \frac{q_1^2}{2\frac{k^{(1)}}{b_{111}^{(1)}} + q_1^2} \tag{4-28}$$

$$\eta \propto \gamma \frac{q_1^2}{\alpha + q_1^2} \tag{4-29}$$

This implies that for low amplitude ($q_1^2 \ll \alpha$), the loss factor is proportional to q_1^2 (which is proportional to the linear energy), which is in accordance to the behaviour reported in literature (section 4-4-1):

$$\eta(q_1^2 \ll \alpha) \propto q_1^2 \propto E_{\mathrm{L}} \tag{4-30}$$

However, for large amplitude ($q_1^2 \gg \alpha$), the q_1^2 dependence vanishes, as the loss factor reaches an upper bound set by γ:

$$\eta(q_1^2 \gg \alpha) \propto \gamma \tag{4-31}$$

This was not reported in literature yet (note that in [4], only small energy levels are considered). Therefore, in the next section simulations are performed to investigate whether this upper bound really exists or not.

4-4-3 Simulations

In this section, the influence of the total energy is investigated by performing numerical simulations (as explained in section 4-2). Simulations are run for increasing initial energy levels at the start. However, the ratio between the energy in the fundamental mode and the total energy (χ) at the start is kept constant. For each set belonging to the same energy level, the time-averaged (η_{ta}) and instantaneous (η_{ins}) loss factor are calculated.

The total energy can be normalised with respect to the pre-strain energy E_{pre} (indicating the nonlinearity reached):

$$\tilde{E}_{\text{t}} = \frac{E_{\text{tot}}}{E_{\text{pre}}} \tag{4-32}$$

The results using this scaling are shown in Figure 4-4. Alternatively, the energy level can be expressed as the equivalent noise temperature (already defined in section 3-4):

$$E_{\text{noise}} = E_{\text{tot}} - E_{\text{noise}} \tag{4-33}$$

$$T_{\text{noise}} = \frac{E_{\text{noise}}}{n_{\text{mode}} k_{\text{B}}}; \tag{4-34}$$

which is shown in Figure 4-7.

Note that because the amplitude of vibration of the fundamental mode becomes quite high, also it's frequency of vibrations increases. To ensure that the eigenfrequencies of the noise modes cover the frequency range of the fundamental mode, only 9 axissymmetric modes are included in the ROM. Those modes are equally spaced in the spectrum (with constant distance of about $1.3 \cdot \omega_1$), and cover a sufficient range of eigenfrequencies (listed in Table 4-3).

Table 4-5: Simulation settings energy scaling. This ROM consist only of axisymmetric modes.

Parameter	Value	Description
n_{modes}	9	Number of modes in the reduced order model
ζ	5×10^{-11}	Viscous damping active on all modes
n_{r}	80	Number of simulations per parameter set
n_{per}	40	Number of periods simulated
χ	0.75	Fraction of initial energy in fundamental mode
$\tilde{V}_0$	0.1-70	Relative initial velocity of the first mode

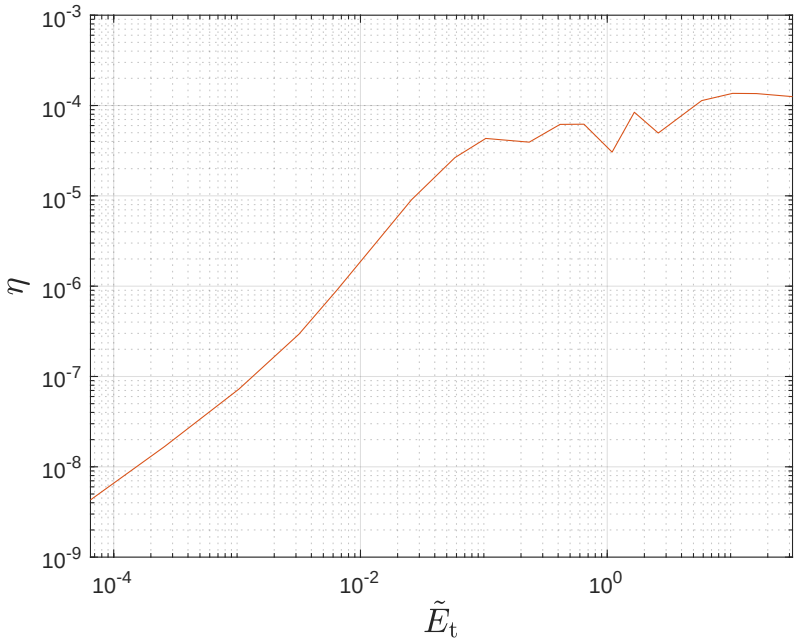


Figure 4-4: Time-averaged loss factor η_{ta} (defined in Eq. (4-19)) versus initial total relative energy $\tilde{E}_{\mathrm{t}}$.

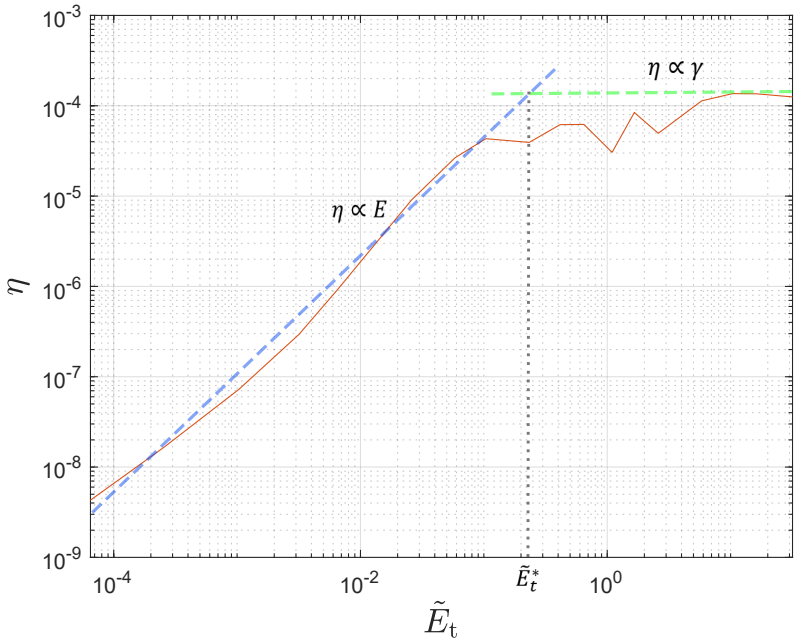


Figure 4-5: Time-averaged loss factor η_{ta} versus initial total relative energy $\tilde{E}_{\mathrm{t}}$, including the asymptotes. The blue line is related to small energy: $\eta \propto E_{\mathrm{L}}$ (Eq. (4-30)). The green line is related to large energy: $\eta \propto \gamma$ (Eq. (4-31)). The transition is located at E^*, which is related to $q_1^2 \approx \alpha$.

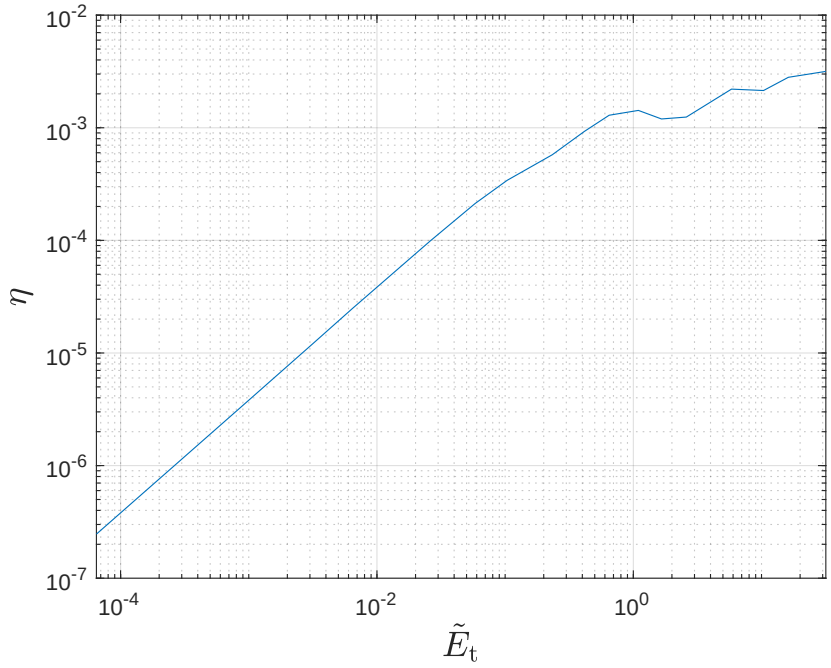


Figure 4-6: Instantaneous loss factor η_{ins} (defined in Eq. (4-21)) versus initial total relative energy $\tilde{E}_{\mathrm{t}}$.

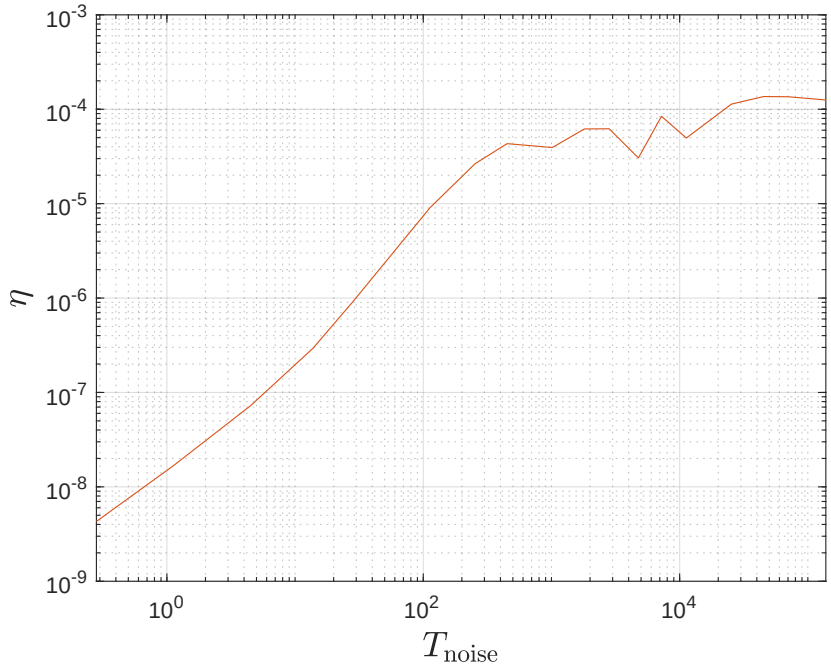


Figure 4-7: Time-averaged loss factor η_{ta} (defined in Eq. (4-19)) versus equivalent noise temperature T_{noise}.

4-4-4 Conclusions

In the simulations, the behaviour predicted in section 4-4-2 indeed appears. For low energy, the loss factor linear increases with energy ($\eta \propto \frac{\gamma}{\alpha} q_1^2$). However, at a certain energy level this relation saturates, as the loss factor reaches an upper bound ($\eta \propto \gamma$).

4-5 Temperature dependence

In the previous section, an equation was derived for the scaling of the loss-factor, whereafter the energy dependence was investigated. However, from the equation (Eq. (4-29)), the loss factor also appears to scale with γ, which depends on the amplitude of the noise modes. In a physical system, the noise energy is set by its temperature. Therefore, in this section the influence of temperature is investigated.

In the next subsection (4-5-1), some work in literature related to this subject is discussed. In section 4-5-2, the behaviour is predicted (using the equations from section 4-4-2), which is validated through simulations in section 4-5-3.

4-5-1 Literature

The fluctuation-dissipation theorem states that, if there is a process that dissipates energy, turning it into heat, there is a reverse process related to thermal fluctuations [24]. As those two phenomena are related to each other, increasing the noise temperature is expected to yield higher damping.

4-5-2 Theory

Recall the energy scaling for the loss-factor, Eq. (4-29)

$$\eta \propto \gamma \frac{q_1^2}{\alpha + q_1^2}$$

where γ was defined in Eq. (4-26) as:

$$\gamma = \frac{E_{\mathrm{NLC}}}{E_{\mathrm{NLS}}^{(1)}} = \frac{\frac{1}{3} b_{112}^{(1)} q_1^3 q_2 + \frac{1}{2} b_{122}^{(1)} q_1^2 q_2^2 + b_{222}^{(1)} q_1 q_2^3}{\frac{1}{4} b_{111}^{(1)} q_1^4}$$

From Eq. (4-26), it becomes clear that the loss-factor depends also on q_2, which is related to the noise energy level (or equivalent noise temperature T). According to the equipartition theorem and assuming the other modes are in thermal equilibrium at temperature T, the noise energy and equivalent temperature was defined in Eq. (4-34) as:

$$E_{\mathrm{noise}} = E_{\mathrm{tot}} - E_{\mathrm{noise}} = n_{\mathrm{mode}} k_{\mathrm{B}} T_{\mathrm{noise}}$$

As the noise energy is expected to be much lower than the signal energy ($E_1 \gg E_{\text{noise}}$), this also holds for the mode amplitudes ($q_1 \gg q_2$). From this, it is expected that for very low noise energy, the loss-factor mainly linearly depends on q_2, which is related to $\sqrt{T}$. For increasing noise energy, the loss-factor scales quadratic ($\gamma \propto q_2^2 \propto T$) or at maximum cubic ($\gamma \propto q_2^3 \propto T^{1.5}$).

4-5-3 Simulations

In this section, simulations are run to investigate the influence of the noise temperature T on the loss factor. To do so, the noise energy is increased for different sets, while the signal energy (energy in mode 1) is kept constant. Simulation settings are listed in Table 4-6, the result is shown in Figure 4-8 and 4-9.

Table 4-6: Simulation settings temperature scaling. This ROM consist only of axisymmetric modes.

Parameter	Value	Description
n_{modes}	9	Number of modes in the reduced order model
ζ	5×10^{-11}	Viscous damping active on all modes
n_{r}	80	Number of simulations per parameter set
n_{per}	50	Number of periods simulated
T_{noise}	$4 - 800\,\text{K}$	Fraction of initial energy in fundamental mode
$\tilde{V}_0$	5	Relative initial velocity of the first mode

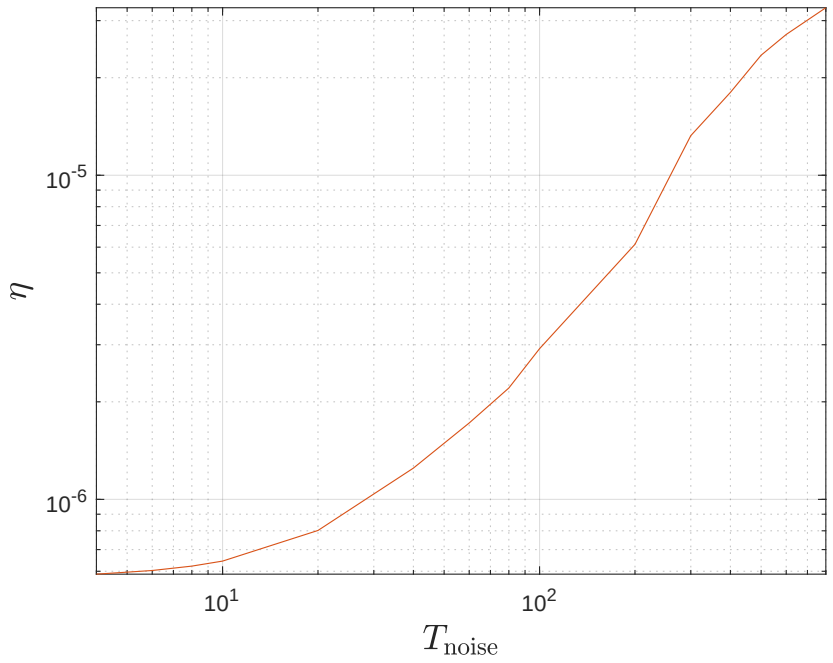


Figure 4-8: Time-averaged loss factor η_{ta} versus equivalent noise temperature T_{noise}.

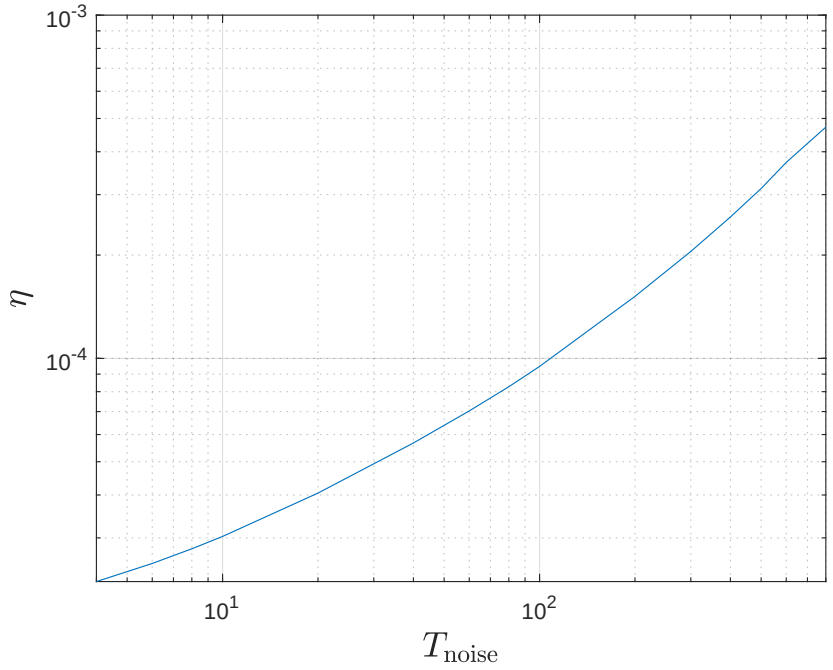


Figure 4-9: Instantaneous loss factor η_{ins} versus equivalent noise temperature T_{noise}.

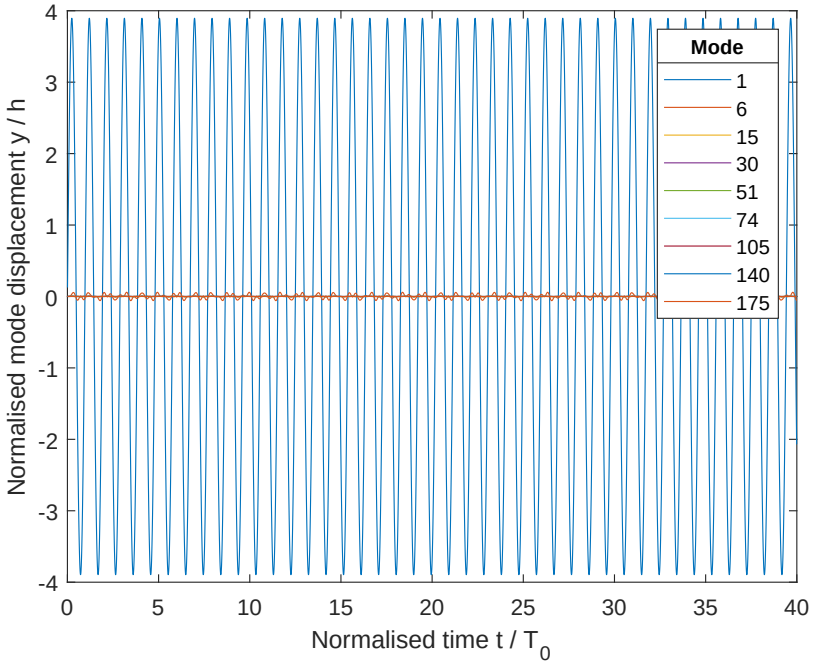


Figure 4-10: Modal displacement for $T_{\text{noise}} = 0$

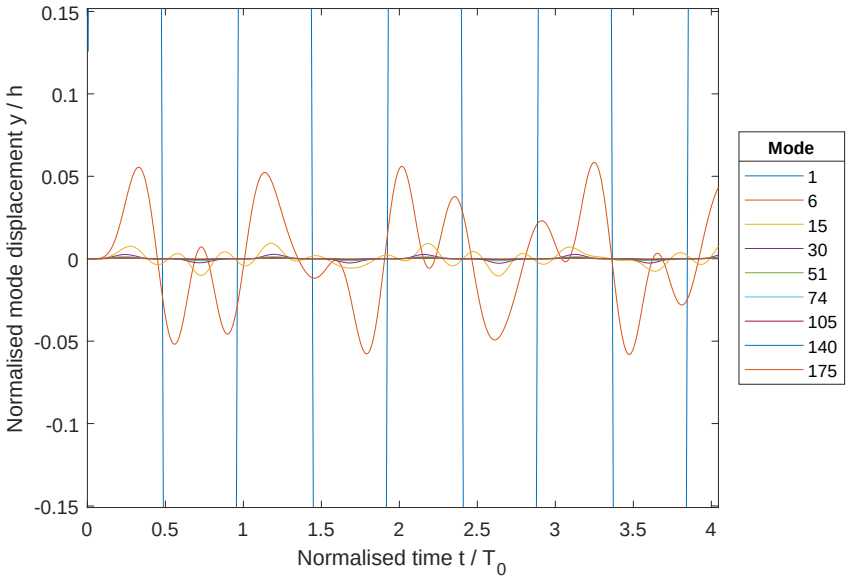


Figure 4-11: Modal displacement, zoomed in around the origin in Figure 4-10 for $T_{\text{noise}} = 0$

4-5-4 Conclusions

In Figure 4-8, indeed an increasing loss factor for increasing temperature can be observed, as was predicted by Eq. (4-26). However, for high temperature, a decrease in scaling is visible. Probably, this is because the signal and noise energy levels are approaching each other, according to Eq. (4-11). For very low temperature (even for $T_{\text{noise}} = 0$, as shown in Figure 4-10 and Figure 4-11), there seems to be a lower bound on the loss factor. Probably this is due to FPU oscillations as described in section 3-4, which can be ignited by the $b_{111}^{(r\neq 1)} q_1^3$ terms in the equations of motion. Apparently, this mechanism transfers a vast energy portion irrespective of the noise energy level. In conclusion, it can be stated that Eq. (4-26) captures the mode-coupling behaviour reasonably, but for very low or high noise temperature, FPU and entropy effects have to be taken into account additionally.

4-6 Membrane diameter scaling

In the preceding sections, the influence of the initial conditions on the loss-factor have been investigated. However, the next sections focus on the influence of the structure itself. This section investigates the loss-factor as a function of the membrane radius, the next section treats nonsymmetric membranes.

In section 4-6-1, some relevant literature is discussed, followed by section 4-6-2, in which theoretical background is provided. Section 4-6-3 includes simulations to test the theory, after which both are compared in section 4-6-4.

4-6-1 Literature

In literature, there are some works concerning the influence of geometry on damping, although the mechanism of mode-coupling damping is not mentioned explicitly. One work which is particularly of our interest was published by R. Barton et al [23]. In their work, the damping of graphene membranes is experimentally determined for different diameters. They found the quality factor to be strongly size-dependent, but could not indicate a reason for this. Their main result is shown in Figure 4-12, where the red line indicates a fit of $Q \sim D^{1.1}$. Further note that in [4], the Q-factor was stated to be radius independent, although no theoretical reasoning was given.

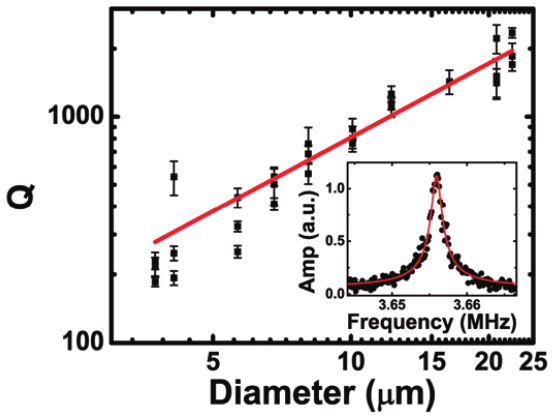


Figure 4-12: Experimentally obtained Q-factor for different membrane diameters, published in [23]. The Q-factor is determined by the half-power bandwidth of each resonance peak. The red line indicates a fit of $Q \sim D^{1.1}$

4-6-2 Theory

From literature, there seems to be evidence for diameter dependent damping in membranes. Therefore, in this section the theory regarding mode-coupling damping is judged with respect to diameter scaling.

As a starting point, consider the loss-factor estimate from Eq. (4-29), derived in section 4-4-2:

$$\eta \propto \gamma \frac{q_1^2}{\alpha + q_1^2}$$

where α is defined as

$$\alpha = 2\frac{k^{(1)}}{b_{111}^{(1)}} \tag{4-35}$$

Both the linear stiffness $k^{(1)}$ and the cubic stiffness $b_{111}^{(1)}$ are structure dependent. Therefore, it's worth taking a closer look at their ratio. The scaling of the linear stiffness of a membrane was written in Table 3-12:

$$k \propto T_0 = \sigma_{\text{pre}} h = \varepsilon_{\text{pre}} E h \tag{4-36}$$

In the same table, the scaling for the nonlinear cubic stiffness was written:

$$b \propto \frac{Eh}{R^2} \tag{4-37}$$

As α was defined as the ratio between those, we find (for a membrane):

$$\alpha = 2\frac{k^{(1)}}{b^{(1)}_{111}} \propto \frac{R^2}{Eh}\varepsilon_{\text{pre}} E h = R^2 \varepsilon_{\text{pre}} \tag{4-38}$$

Recall that for small vibrations, the loss-factor is inversely proportional to α:

$$\eta(q_1 \ll \alpha) \propto \frac{\gamma}{\alpha} q_1^2 \tag{4-39}$$

Fixing the other parameters yields:

$$\eta \propto \frac{1}{\alpha} = \frac{1}{R^2 \varepsilon_{\text{pre}}} \tag{4-40}$$

and for the Q-factor (defined as the inverse):

$$Q = \frac{1}{\eta} \propto R^2 \varepsilon_{\text{pre}} \tag{4-41}$$

This result resembles the experimental results from literature (Figure 4-12), although the relation is stronger (quadratic) than yielding from the data fit. Other damping sources could also play a role in determining Q, like edge losses and material losses. Moreover, it would be of interest to test the linear dependence on prestrain epsilon.

Effective Q-factor

In practice, the fundamental mode can have damping itself due to other damping effects than mode coupling. Consider each damping effect to result in an energy loss ΔU_i every period:

$$\eta_{\text{tot}} = \frac{\Delta U_1 + \Delta U_2 + ...}{2\pi U} = \frac{\Delta U_1}{2\pi U} + \frac{\Delta U_2}{2\pi U} + ... = \eta_1 + \eta_2 + ... \tag{4-42}$$

which influences the resulting Q-factor in a nonlinear way:

$$Q_1 = \frac{1}{\eta_{\text{tot}}} = \frac{1}{\eta_{\text{s}} + \eta_{\text{c}}} \tag{4-43}$$

Here the indices represent solo (η_{s}, representing a constant intrinsic damping mechanism) and coupling (η_{c}) damping. The resulting Q-factor can therefore have a weaker dependence (depending on the ratio between the coupling and solo damping) on the physical quantities, compared to the case of no intrinsic damping in the fundamental mode.

4-6-3 Simulations

In the previous section, it was derived that the loss-factor is expected to scale inversely to the radius squared. Therefore, in this section, this hypothesis is tested using simulations. The structure used for experiments in [23] is modelled, as described in section 4-3. Using constant initial noise and signal energy, simulations are run for different membrane radii, after which the loss-factor is extracted. Simulation settings are listed in Table 4-7, the results are shown in Figure 4-13 and 4-14. In Figure 4-17, the amplitude signal of one of the simulations is given.

Table 4-7: Simulation settings radius scaling. This ROM consist of the first 12 unique modes, listed in Table 4-4

Parameter	Value	Description
n_{modes}	12	Number of modes in the reduced order model
ζ	5×10^{-11}	Viscous damping active on all modes
n_{r}	100	Number of simulations per parameter set
n_{per}	50	Number of periods simulated
χ	0.5	Fraction of initial energy in fundamental mode
$\tilde{V}_0$	30	Relative initial velocity of the first mode
$\tilde{E}_T$	8.7	Relative total energy

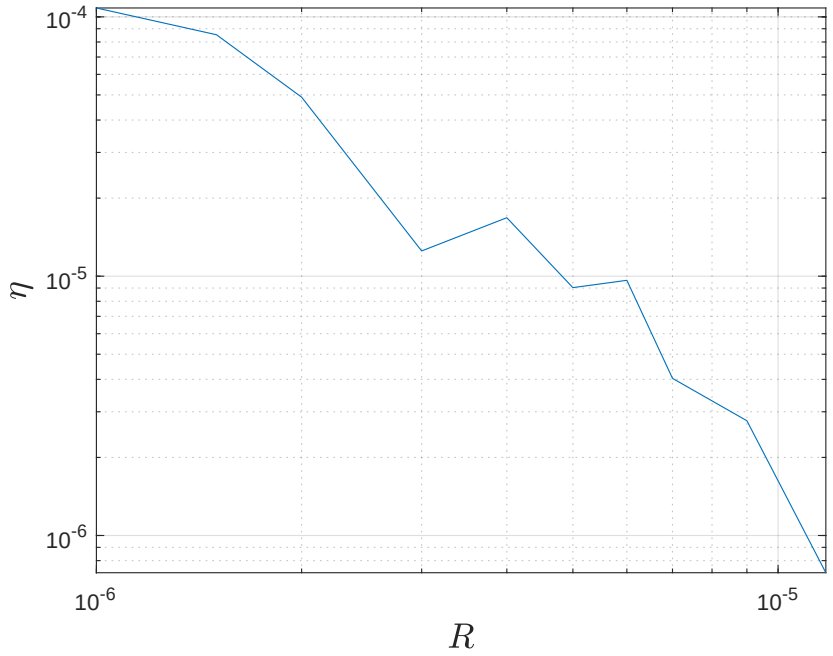


Figure 4-13: Time-averaged loss factor η_{ta} for varying radius.

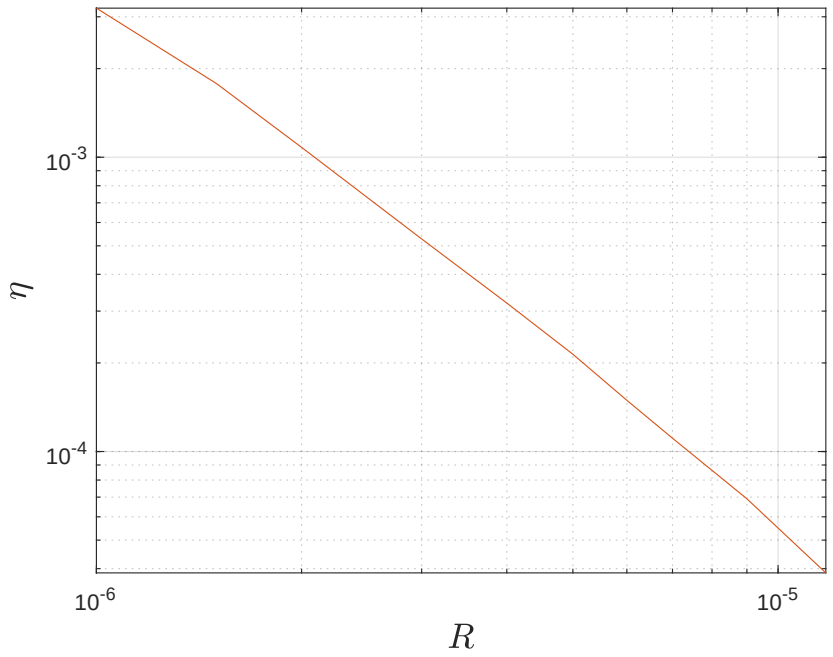


Figure 4-14: Instantaneous loss factor η_{ins} for varying radius.

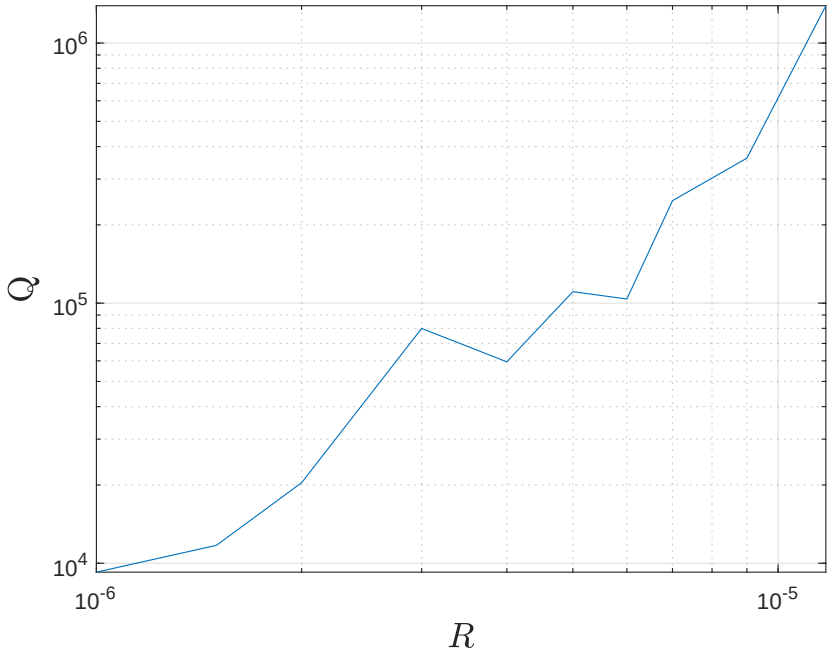


Figure 4-15: Time-averaged Q factor for varying radius.

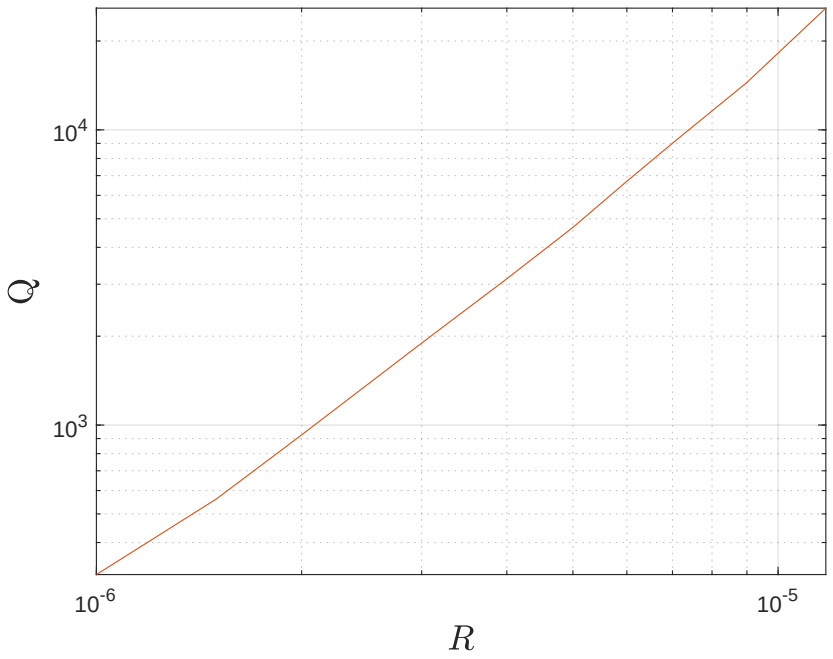


Figure 4-16: Instantaneous Q factor for varying radius.

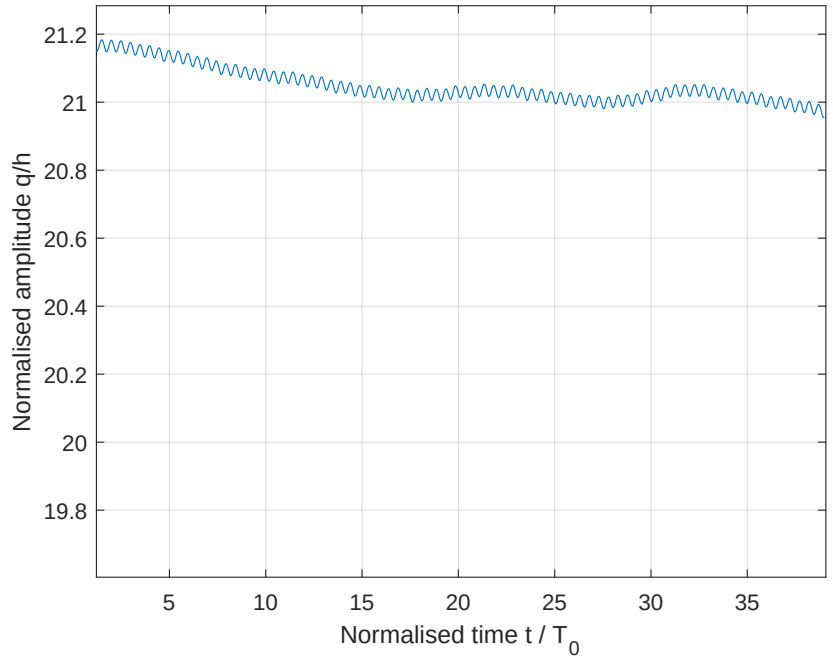


Figure 4-17: Amplitude signal of one (arbitrary) simulation run.

4-6-4 Conclusions

From Figure 4-13, the radius dependence can be approximated as $\eta_{\text{ta}} \sim R^{-2.0}$ and $\eta_{\text{ins}} \sim R^{-1.7}$. The time-averaged loss factor η_{ta} is in good accordance to the theoretical prediction, although it is fluctuating slightly. For small radii, the effect seems to decrease slightly. Possibly, the amplitude of vibration is reaching the saturation regime; ($q_1^2 \sim \alpha$), so the influence of α decreases, reducing the effect of the radius. Regarding to experimental data discussed in 4-6-1, probably the intrinsic damping in the fundamental mode plays a role as depicted in the end of section 4-6-2.

Concluding, it can be stated that mode coupling damping can indeed be a source of diameter dependent damping, and could be a candidate for the damping behaviour observed in experiments reported in literature.

4-7 Elliptic membranes

This section treats the influence of vibration frequency (and thus indirectly internal resonance) on mode coupling damping. To do so, the simple circular membrane which was used so far is changed to an oval by altering the aspect ratio. This changes the eigenfrequency spectrum gradually, allowing for a systematic investigation of the effect on the loss-factor when frequency differences between coupled modes change. In section 4-7-1, relevant literature is shown. Subsequently, section 4-7-2 provides a theoretical background, followed by simulations in section 4-7-3 which are discussed in the end.

4-7-1 Literature

For a circular membrane, the eigenfrequencies are located at a fixed value with respect to the fundamental eigenfrequency. When neglecting bending stiffness, the spectrum is independent of any material (density, Young's modulus) or geometric (thickness, radius) properties. However, by making the membrane elliptic, the eigenfrequencies shift with respect to each other (the same can be obtained by altering the pre-stress ratio). The eigenfrequencies can be determined analytically using Mathieu functions, which is shown in [25].

Several works treat the shape-dependence of damping in mechanical structures. However, here mainly mechanical damping mechanisms like anchor losses are subject of investigation [26]. Literature or experiments concerning the influence of a structure's shape on mode-coupling damping is unknown to the author.

4-7-2 Theory

In section 4-1-2, Eq. (4-8) was derived, revealing the energy transfer between modes to be a function of their relative vibration frequency:

$$W_{112}^{(1)} = b_{112}^{(1)} \hat{q}_1^3 \hat{q}_2 \frac{1}{8} \left(\frac{\cos(2\pi m\delta)}{\delta} - \frac{\cos(2\pi m\delta_3)}{\delta_3} - \frac{\cos(2\pi m\sigma)}{\sigma} + \frac{\cos(2\pi m\sigma_3)}{\sigma_3} \right) \tag{4-8}$$

or

$$W_{112}^{(1)} = cE_{1112} f(\omega_1, \omega_2) \tag{4-10}$$

In Figure 4-18, the energy dependent function $f(\omega_1, \omega_2)$ is shown. At $\frac{\omega_2}{\omega_1} = 1$ or 3 (the conditions for 1:1 of 1:3 internal resonance), the work ($W_{112}^{(1)}$) becomes infinite. Therefore, modes with frequency ratio's close to this can show strong interaction. Energy transmission vanishes for modes with eigenfrequencies far apart (from about $\frac{\omega_2}{\omega_1} > 5$).

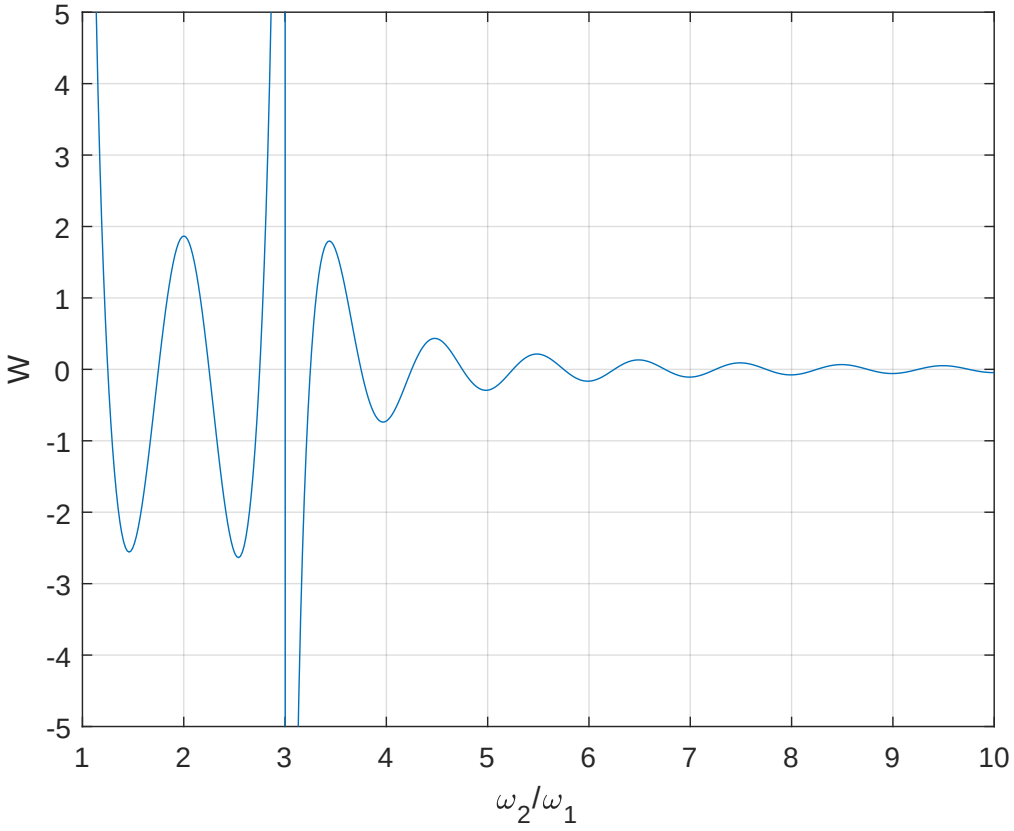


Figure 4-18: $f(\omega_1, \omega_2)$: Frequency dependence of the work of mode 1 on the cubic spring.

Note that for cubic stiffness terms with other index repetition (like, when 3 modes are included, $b_{123}^{(1)} q_1^2 q_2 q_3$), the frequency dependence may change. However, the fluctuating behaviour including sharp peaks can be seen as characteristic for the frequency dependence.

4-7-3 Simulations

For oval membranes, eigenfrequencies change and degenerate modes are not present anymore. In Figure 4-19, the eigenfrequencies of the first 20 modes (normalised to the first eigenfrequency) are shown for different aspect ratio's. Note that altering the pretension in one direction has the same effect (compared to the radius, it scales with square root).

From Figure 4-19, it becomes clear that the eigenfrequencies come closer to the fundamental eigenfrequency for higher aspect ratio (the spectrum becomes denser in this area). According to (4-8), power transfer increases when vibration frequencies approach each other. However, mode interaction can occur at many different frequency ratio's of multiple modes, and is difficult to predict. Therefore, in this section, we change the aspect ratio to investigate the influence of frequency spacing between the eigenmodes on the loss factor.

In Table 4-8, the properties of the simulated membrane are listed, and Table 4-9 shows the used simulation settings. The results are shown in Figure 4-21 and 4-20.

Table 4-8: Structure parameters of an oval membrane

Parameter	Value	Description
E	1044 GPa	Youngs modulus monolayer graphene
ν	0.16	Poisson's ratio
T_0	$0.035\,\mathrm{N\,m^{-1}}$	Pretension
h	0.335 nm	Membrane thickness
R_1	300 nm	Radius

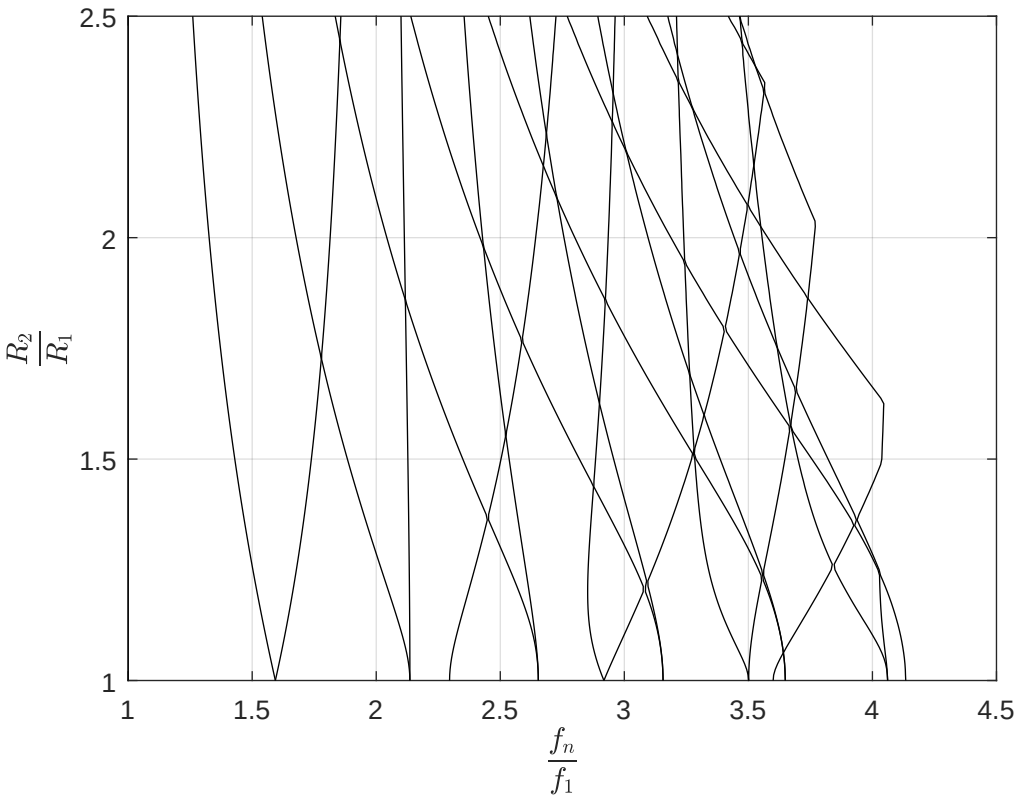


Figure 4-19: Normalised eigenfrequencyversus aspect ratio for 20 modes

Table 4-9: Simulation settings to estimate the loss-factor of an oval membrane.

Parameter	Value	Description
n_{mode}	20	Number of modes in the reduced order model
ζ	5×10^{-10}	Viscous damping active on all modes
n_{per}	20	number of periods simulated
n_{r}	400	number of simulations per batch
χ	0.3	Fraction of initial energy in fundamental mode
$\tilde{V}_0$	4	Relative initial velocity: $\tilde{V}_0 = \frac{V_0}{\omega_1 h}$

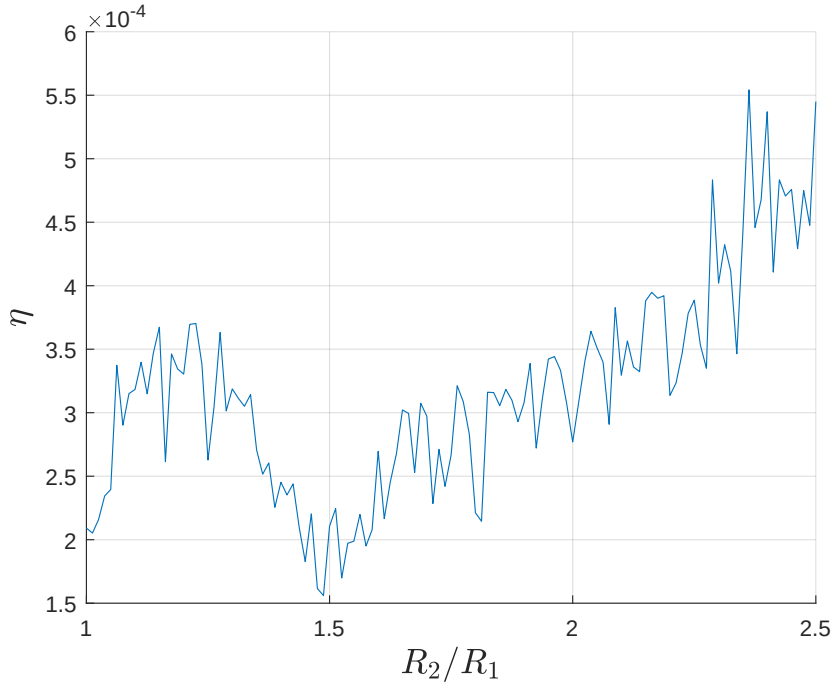


Figure 4-20: Time-averaged loss factor η_{ta} of an oval membrane for different aspect ratio's.

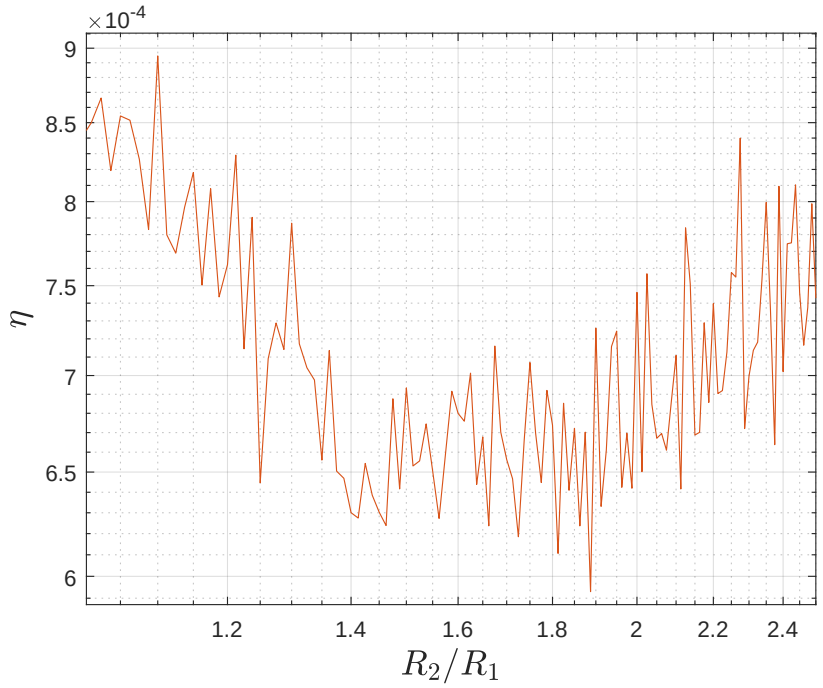


Figure 4-21: Instantaneous loss factor η_{ins} of an oval membrane for different aspect ratio's.

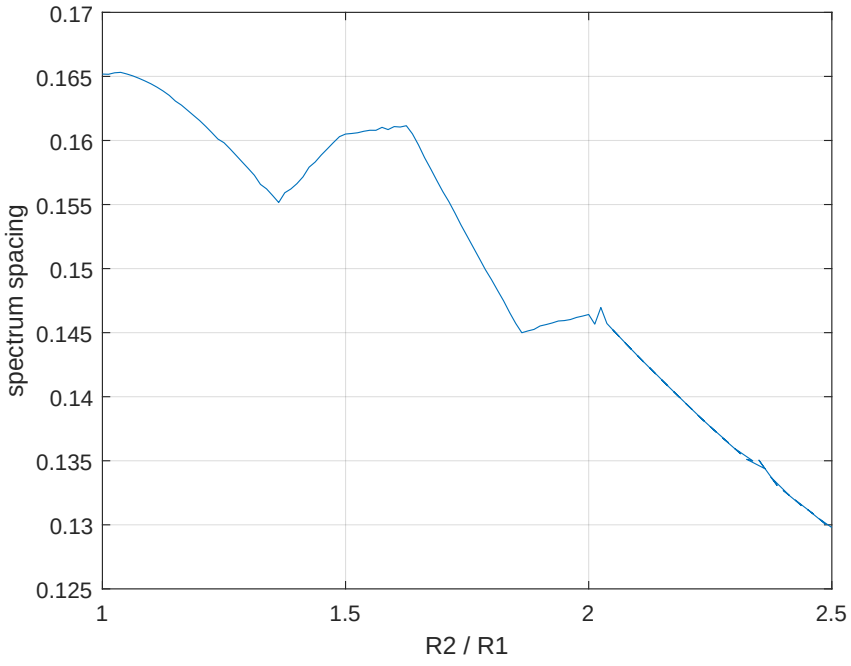


Figure 4-22: Spectrum spacing of an elliptic membrane for different aspect ratio's.

The spectrum spacing can be derived from the distance between the eigenfrequencies:

$$d_1 = \omega_2 - \omega_2, \qquad d_2 = \omega_3 - \omega_2, ... \tag{4-44}$$

Then, the spectrum spacing (normalised with respect to the fundamental eigenfrequency) of the structure is defined as follows:

$$\Upsilon = \frac{1}{N\omega_1}\sqrt{\sum_{i=1}^{N} d_i^2} \tag{4-45}$$

Here, N equals the number of distances d_i (one less than the considered number of eigenfrequencies). A small spacing implies that the eigenfrequencies are almost equally distributed, whereas a large spacing implies large gaps between them.

Further note that in this work, only the energy flow from the fundamental mode is considered. However, it could be interesting to investigate the energy flow between higher modes, as for some aspect ratio's, those modes have equal eigenfrequency (intersections in Figure 4-19).

4-7-4 Conclusions

In conclusion, the eigenfrequency spectrum is found to have significant influence on the loss-factor, although the variation remains within one order of magnitude. Part of the aspect ratio dependence can be predicted from the eigenfrequency spectrum, but on detail level, only simulations reveal the true behaviour. In general, it can be recommended to stick to a circular shape or an aspect ratio of 1.5 to minimize mode-coupling damping.

Chapter 5

Discussion

This chapter contains a discussion concerning the presented work. Some of the topics are further elaborated in the recommendations (chapter 7).

5-1 Methods

In chapter 2, the STEP method was found to be most suitable for our purpose. In the following, the method is discussed with respect to applicability, computational cost and accuracy.

5-1-1 In-plane STEP

The in-plane STEP variant can yield a very complete description of the dynamics of a structure, It can be seen as the only way to 'exactly' model in-plane inertia and damping. At least for simple structures, the effect of in-plane dynamics can be investigated thoroughly, without the need to introduce artificial terms like $x^2\ddot{x}$ & $x\dot{x}^2$ (for inertia) or $x^2\dot{x}$ (for damping).

However, the method is only feasible as long as in-plane motion can sufficiently be represented by a limited number of eigenvectors. For plane-stretching nonlinearity (like strings and membranes have), this is the case, but for bending nonlinearity (like beams have), many eigenvectors are needed. If a lot of in-plane eigenvectors are needed, the ROM becomes bulky, strongly increasing computational cost of the STEP process as well as all post-processing. A second drawback of this method can be found in the fact that in-plane modes usually have a much higher eigenfrequency. Therefore, they require much finer time-discretization for post-processing, increasing computational cost.

Last, the selection procedure for in-plane modes is only validated for 2D structures, for which it worked fine. However, as it searches for strong quadratic coupling stiffness, selected modes does not necessarily need to be 'in-plane', but could also be of different nature. Nevertheless, as the selected softening mode is strongly coupled, and thus significantly influences the corresponding out-of-plane mode, it should be included for the same reason we include other in-plane modes.

5-1-2 Condensed STEP

If in-plane dynamics can be neglected, for 2D structures the condensed STEP variant is proposed (section 2-4-2). For most structures analysed in this work, this method was used. It was observed that for analysing fixed-free structures like a cantilever, it took the static nonlinear solver more effort to meet set tolerances. First, this could be due to the fact that bending nonlinearity is harder to detect as it arises at larger deformation compared to plane stretching nonlinearity. Second, its nature is of higher order than cubic, making the result depending on the amplitude of the prescribed displacements. Third, calculated force fields show a force concentration at the free tip, probably indicating that Comsol has difficulties finding a balanced force distribution.

5-1-3 STEP - performance

Concerning the accuracy of the STEP method, it may be observed that the coefficients are sequentially calculated [7]. As a result, errors are accumulated in the last computation step concerning $b^r_{i\neq j\neq k}$ coefficients. This could be improved by writing all equations in one system and additionally including the equality relations between them. In this way, the solver can search for the solution yielding the smallest error. To give an indication of the current accuracy, some key values are listed in Table 5-1.

Computation time

Because the determination of stiffness coefficients relies on solving a full nonlinear static finite element problem, it can be regarded computationally more expensive than for example analytic methods. However, this does not have to be a serious problem. First of all, when judging a structures performance in practice, it is subjected to numerous loadcases related to it's functionality. Therefore, loadcases related to nonlinear dynamics using the STEP method should in practice not significantly increase the total computational cost.

For optimization purposes, where the structure's dynamic behaviour have to be evaluated for many different designs, the STEP method might be less suitable (especially for heavy structures & a large number of included eigenmodes). Note that this doesn't hold for the case in which only a subset of coefficients is optimized, as the determination of a subset can be done much faster.

Nevertheless, for the structures analysed in this work, the processing time of the STEP method usually was comparable or less than the time needed for post-processing (like bifurcation analysis). Some typical values indicating the performance are listed in Table 5-1.

5-2 Dynamics

In chapter 3, nonlinear effects in simple structures were investigated. For all structures, hardening was observed, but other effects like internal resonance only showed up in a few cases. However, if for a structure a certain effect is not observed in this work, this does not imply that it cannot happen. For every analysis, choices had to be made regarding simulation

Table 5-1: Testcase STEP method for a circular membrane, modelled using 3D membrane elements in Comsol. Physical properties are listed in Table 4-8. The relative error is defined in appendix C-2-2

Value	Description
$20 \cdot h$	prescribed displacement amplitude (h equals thickness)
2619	number of dof
20	number of modes in the reduced order model
1750	number of loadcases
1606 sec (27min)	total computation time
0.058	maximum relative error
0.0054	mean relative error

settings like damping, force level and excitation point. Therefore, using different conditions, perhaps effects could be discovered which remained unrevealed in this work.

In section 3-3-2, simulations were performed to mimic experimental results. While the resemblance was quite good, still some physical mechanisms behind the experimentally observed effects remain unknown. For example, the mechanism creating parametric excitation could be further investigated using multiphysics modelling software. In this way the parametric excitation mapping for each mode can be determined from first principle. When the parametric modal mapping is known, modal damping can be determined more certainly using the period doubling bifurcation threshold (as depicted in section 3-2-3).

5-3 Quality factor

In chapter 4, mode coupling damping was investigated by simulating free vibrations. This method was regarded as the most 'clean' approach to capture the effect, as no other damping sources are present in the analysis. In this way, all observed damping is definitely due to mode-coupling. In section 4-2, it was already shown that the damping behaviour is quite difficult to extract, as the modes show slow oscillations in energy level, on top of a gradual decay. By defining two loss factors, both effects are captured separately.

However, there are some settings which may possibly have influence on the result. The number of simulated periods (n_{per}) for example can have slight influence depending on the energy regime. It should be long enough to ensure the modes can develop interaction in order to exchange energy. On the other hand, it should remain short to stay close to the initial energy levels. As the observed Q-factor was throughout all simulations larger than about 1000, the second condition is assumed to be met for all simulations. For the sake of computational cost, n_{per} is kept in the range of 20-80 (depending on the number of included modes).

Another difficulty arises from the energy equations. The equations describing coupling energy are costly to evaluate for large systems, as the number of terms increases rapidly. Therefore, it is not practical to monitor the total energy in the system for all simulations. Alternatively, for some typical scenario's the numerical accuracy is investigated (appendix C-1-1) by considering the total system energy, yielding proper simulation settings to be used subsequently.

Looking at the results, it may be noticed that the time-averaged loss factor shows quite some wobbling behaviour. Because the system approaches chaos for large energy levels, the response can be completely different for slightly changed initial conditions (it's ill-posed). To account for this, a large number of initial conditions is simulated (section 4-2), of which statistical values are extracted. The result is a trade-off between computational cost (increasing with the number of simulations n_r) and statistical coherence.

Furthermore, it should be noted that the experimental data mentioned in section 4-6-1 concerns forced vibration. Here, the quality factor is determined from the half-width power, which might be influenced by noise (altering the pretension), pretending higher damping. For a good comparison, free-vibration experiments should be done, varying the important parameters mentioned in chapter 4.

On the other hand, also forced vibration could be studied numerically regarding mode coupling damping. However, as this concerns periodic (steady state) solutions, there is way less freedom in setting initial settings like energy and noise. Moreover, some means of damping should be added to ensure bounded solutions, which could trouble the mode coupling damping measurement.

5-3-1 Mode coupling damping in graphene

As mentioned in the introduction, the performance of graphene sensors is currently strongly limited by it's low quality factor. From experiments (section 3-3-2), the quality factor of a multilayer graphene membrane was found to be approximate 180, and 300-2000 for a monolayer membrane (section 4-6-1). In both multi and monolayer graphene, this low quality factor was already in the linear regime experimentally observed. However, the smallest observed (time-averaged) quality factor in simulations is about 10000, occurring far in the nonlinear regime. Therefore it can be concluded that mode-coupling is not the dominant damping mechanism in graphene.

However, for large vibration amplitude, mode coupling damping can be very important. From section 3-3-3, it is found that the parametric resonance curve can be blocked when there is strong interaction with other modes. As their amplitude increases, they dissipate more energy (due to their viscous damping) which they tap from the fundamental mode. This process can also be seen as mode coupling damping. In full agreement with section 4-4, this effect only occurs at large vibration amplitude.

Second, in section 3-3-1 it is shown that the resonance curve of the fundamental mode can be significantly flattened due to the presence of higher modes. Similar to the shooting phenomena, the flattening is caused by the rise of higher modes, resulting in mode coupling damping of the fundamental mode. Again, this phenomena occurs at high vibration amplitude, confirming section 4-4. Although this analysis concerns a silicon membrane having very low intrinsic damping, the phenomena of very rapidly rising higher modes (or internal resonance) could occur in any structure.

Further note that the instantaneous loss factor (defined in section 4-2-2) indicates the capability of power transmission between modes rather than a general decay rate. This value is usually an order of magnitude larger than the time-averaged loss factor, and occurs fewer. Therefore, it shows resemblance to forced vibrations, where at some scarce conditions, strong coupling appeared (recall for example the shooting effect in section 3-3-3).

Chapter 6

Conclusions

In this chapter, conclusions are made regarding the complete work. For every subsection, research questions are repeated, followed by the answers, yielding from the concerning chapter.

Main conclusion

The research question was defined as follows:

> *How can the nonlinear dynamics of a structure be described in a simple model using Finite Elements, and to what extend do those nonlinearities limit the quality factor?*

Using the condensed STEP method, a compact nonlinear ROM can be constructed, which is perfectly suited for further dynamic analysis. The condensed STEP method is an extension of the original STEP method and was developed in this project. It was implemented for 2D structures, and appeared to be accurate and easily applicable for different types of elements. Through a connection to the powerful software AUTO, the dynamic behaviour can be simulated. In this way, the nonlinear ROM is able to describe experimentally observed complicated effects like parametric resonance and shooting. Due to nonlinearities, energy flows from highly energetic modes to calmer ones, limiting the quality factor of an excited mode. Typically, the quality factor due to mode coupling is larger than 1×10^4, although for short time-spans, it may drop to about 1×10^3. The damping process is described by several scaling laws, giving opportunities to reduce the damping effect and design for better performance.

Methods

How can a simple mathematical model be built to describe the nonlinear dynamics of a structure using FEM?

Using the condensed STEP method, a compact nonlinear ROM can easily be constructed, implicitly accounting for in-plane motion. For 2D structures, the method proved to be accurate and straightforward applicable for different types of elements. Optionally, in-plane modes can be selected automatically, to include in-plane inertia and damping. Other FEM based methods require in-dept FEM knowledge and have severe difficulties to include in-plane softening, making them less appealing.

Dynamics

How does nonlinear stiffness influence the dynamic behaviour of simple structures?

For straight strings, beams and flat membranes, it was found that cubic stiffness is present, causing hardening of the resonance curve. Moreover, through cubic coupling terms, modes excite each other, supporting modal interaction. For a circular membrane, this may result in significant vibration of higher modes when the fundamental mode is parametrically excited. For a special case, shooting occurs in simulations, strikingly matching experimental results. Concerning free-vibrations, FPU periodicity can be observed due to mode coupling. For a circular membrane, an energy region of stable FPU periodicity exists, while in a string these FPU oscillations appear to be unstable. An overview of all observed dynamic effects is provided in Table 3-39.

Quality factor

How can the energy flow due to nonlinear mode coupling be described?

The energy flow due to nonlinear mode coupling is found to obey certain scaling laws. Mode coupling increases with signal and noise energy, but reduces with radius or pre-stress. These scaling laws have been analytically derived, and are confirmed by simulations. Additionally, the coupling proved to be shape and eigenfrequency spectrum dependent. Globally, the spectrum spacing of an elliptic membrane can be related to mode coupling-damping.

Chapter 7

Recommendations

This chapter contains recommendations concerning future research. Some of them where already partly explored, but not reported as they were not directly connected to the research goals of this work.

Methods

Regarding the current implementation, some improvements could be made. The computational load of the STEP method consists of independent loadcases, which could all be handled in parallel (parallel computing, using multiple cores). This feature could be further exploited in order to reduce the processing time of computational heavy structures. Second, the STEP method could be exploited for multiphysic dynamics (for example a string in a magnetic field).

Third, the condensed STEP method could be used to approximate in-plane damping & inertia around the eigenfrequency of the corresponding out-of-plane mode. Here, in-plane damping and inertia can result in nonlinear damping and nonlinear mass acting on the out-of-plane (OOP) mode. Their influence (expressed in nonlinear damping and mass terms in the equations) on the OOP mode could be determined by prescribing velocity and acceleration instead of displacement fields in the STEP method. Then, the equation of motion of the in-plane mode can be removed, yielding a much smaller ROM. Fourth, the applicability of proposed methods to 3D structures could be investigated, as the compliant STEP method is expected to be applicable without any limitations on structure geometry.

Dynamics

Beside starting new directions of research, some effects shown in this work deserve further investigation. Shooting behaviour was simulated and related to experiments in section 3-3-3. However, this could be made complete by doing a systematic study, investigating both theory (like necessary conditions) and applications (like determination of sample material properties). Second, FPU effects could be further investigated both theoretically (defining key parameters and testing other structures) and experimentally.

Using the described methods, in chapter 3 the dynamics of a couple of structures where analysed. However, this is just a tip of the iceberg, as much more types of structures could be investigated. As a first step close to this work, other geometries like for example triangles and diamonds could be studied (as those have a different eigenspectrum and coupling). Thereafter, the effect of varying thickness throughout the structure could be an interesting follow up.

It can be observed that most analysed structures in chapter 3 have no quadratic nonlinear stiffness. However, by altering geometry (for example introducing asymmetry), quadratic stiffness could arise, inducing dynamic effects like softening. Fourth, using the preceding knowledge, efficient ways of energy transmission between modes would be worth studying. As mode interaction is quite complicated to predict on forehand, (as it depends as well on multiple coupling stiffnesses as on frequency ratio), presented methods can be used to investigate beneficial conditions.

Qfactor

In chapter 4, mode-coupling damping was theoretically investigated. It would be very interesting to compare the discovered scaling laws with experiments (Ideally, using materials having low internal damping.). Second, mode-coupling damping could be investigated for forced vibrations. Here, intrinsic mode damping cannot be ignored and energy flow has to be steady (for stable solutions). Third, the effect of viscous damping on mode coupling damping for free vibration could be investigated, as some literature [20] suggest this to be an important factor. Fourth, another simulation setup could be considered. Instead of starting from randomised initial conditions, a steady state solution found from the frequency sweep could be used. Last, perturbation techniques could be applied to the equations of motion. Possibly, this would yield insight in as well FPU behaviour as mode-coupling damping.

Appendix A

Supplementary material for Chapter 2

In this appendix, supplementary material to chapter 2 can be found.

A-1 Remarks

A-1-1 In-plane mode selection method

Remarks on the selection procedure:

- If no matching in-plane modes can be found, it can be worth refining the mesh, as the higher modes can be sensitive to it.
- For more complicated structures, multiple in-plane mode can have a softening effect on a single out-of-plane mode. They should all be included in the ROM.

A-1-2 Matlab-AUTO interface

In order to study forced vibrations, 'AUTO 97: Continuation and bifurcation software for ordinary differential equations' by E.J. Doedel is used. To compile the Fortran workspace, Visual Fortran Professional Edition 6.1.0 is used. Some notes on the use of AUTO:

- for lower computation time
  - switch bifurcation detection off
  - use low values for discretisation (NTST & NCOL)
  - In the compiler, set optimalization for runtime speed. (at the cost of compilation time). This works for small systems.
- a small amount of damping is needed

- the number of characters per line and the number of line continuations are limited. For long statements, functions could be used.
- output is written realtime, so in case of unexpected long computations, the output files can be consulted.
- make sure to declare all variables before assigning values to them
- to start a nw branch from a bifurcation point, make sure the bifurcation point is found using a very small step size.
- the phase of the excitation (the solutions to the ODE producing sin and cos) can change for different solutions. Make sure the extracted phase of the state vectors are with respect to the phase of the excitation signal
- create an NMAKE file to easily use the API to compile while using preferred compilation settings

A-2 Mathematical derivations

Softening calculation

Consider the static equations coupling two modes:

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + a_{12}^{(1)} q_1 q_2 + a_{22}^{(1)} q_2^2 + b_{111}^{(1)} q_1^3 + b_{112}^{(1)} q_1^2 q_2 + b_{122}^{(1)} q_1 q_2^2 + b_{222}^{(1)} q_2^3 = F^{(1)} \tag{A-1}$$

$$k_2 q_2 + a_{11}^{(2)} q_1^2 + a_{12}^{(2)} q_1 q_2 + a_{22}^{(2)} q_2^2 + b_{111}^{(2)} q_1^3 + b_{112}^{(2)} q_1^2 q_2 + b_{122}^{(2)} q_1 q_2^2 + b_{222}^{(2)} q_2^3 = F^{(2)} \tag{A-2}$$

Only the $a_{11}^{(i)}$ and $b_{111}^{(i)}$ coefficients are known, but they are related to other terms (see appendix C-2-1 Eq. (C-15)). Removing all terms with unknown coefficients (this will be justified later) yields:

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + a_{12}^{(1)} q_1 q_2 + b_{111}^{(1)} q_1^3 + b_{112}^{(1)} q_1^2 q_2 = F^{(1)} \tag{A-3}$$

$$k_2 q_2 + a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3 = F^{(2)} \tag{A-4}$$

For a given q_1, the corresponding q_2 can be calculated (for $F^{(2)} = 0$):

$$q_2 = -\frac{a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3}{k_2} \tag{A-5}$$

Note that the $b_{112}^{(2)} q_1^2 q_2$ term is of equal order in q_2 as $k_2 q_2$, and should actually be be present in the denominator of Eq. (2-19). However, at this point it is still unknown, and is not necessary to estimate the mode softening effect. Eq. (2-19) can be substituted into Eq. (2-17):

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + -a_{12}^{(1)} q_1 \frac{a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3}{k_2} + b_{111}^{(1)} q_1^3 + -b_{112}^{(1)} q_1^2 \frac{a_{11}^{(2)} q_1^2 + b_{111}^{(2)} q_1^3}{k_2} = F^{(1)} \tag{A-6}$$

The $b_{111}^{(2)}$ and $b_{112}^{(1)}$ coefficients correspond to terms of order higher than 3, and thus will be neglected as the original out-of-plane equation does not contain this order of terms in q_1:

$$k_1 q_1 + a_{11}^{(1)} q_1^2 - a_{12}^{(1)} q_1 \frac{a_{11}^{(2)} q_1^2}{k_2} + b_{111}^{(1)} q_1^3 = F^{(1)} \tag{A-7}$$

Use $a_{12}^{(1)} = 2a_{11}^{(2)}$

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + \left(b_{111}^{(1)} - 2\frac{\left(a_{11}^{(2)}\right)^2}{k_2} \right) q_1^3 = F^{(1)} \tag{A-8}$$

$$k_1 q_1 + a_{11}^{(1)} q_1^2 + b_{111}^{(1)} (1 - \vartheta_2^{(1)}) q_1^3 = F^{(1)} \tag{A-9}$$

$$\vartheta_2^{(1)} = 2\frac{\left(a_{11}^{(2)}\right)^2}{b_{111}^{(1)} k_2} \tag{A-10}$$

Now, ϑ can be seen as the relative influence of the in-plane mode i on the out-of-plane mode j, and thus acts as a criteria to determine whether the in-plane mode should be incorporated into the ROM or not. Note that for all modes, the a and b coefficients needed to calculate $\vartheta_i^{(j)}$ for one UOP mode are available from only applying two displacement fields to the structure (the very first stage of the STEP method, Eq. (2-13)).

A-2-1 Time and space scaling

Scaling is applied such that for the first eigenmode in the ROM, unit mass, stiffness and nonlinear stiffness are obtained. Consider the sdof equation for the first mode:

$$m\ddot{u} + k_1 u + k_2 u^2 + k_3 u^3 + c\dot{u} = F\sin(\omega t) \tag{A-11}$$

Dividing by mass m yields

$$\ddot{u} + \frac{k_1}{m} u + \frac{k_2}{m} u^2 + \frac{k_3}{m} u^3 + \frac{c}{m}\dot{u} = \frac{F}{m}\sin(\omega t) \tag{A-12}$$

Which is simplified to:

$$\ddot{u} + ku + au^2 + bu^3 + 2\zeta\omega_n \dot{u} = f\sin(\omega t) \tag{A-13}$$

Space is scaled the following way:

$$u = \tilde{u}h \longrightarrow \tilde{u} = \frac{u}{h} \tag{A-14}$$

Time is scaled as:

$$t = \tau T \longrightarrow \tau = \frac{t}{T} \tag{A-15}$$

Yielding time derivatives:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \frac{\mathrm{d}u}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}t} = \frac{\mathrm{d}u}{\mathrm{d}\tau}\frac{1}{T} \tag{A-16}$$

$$\frac{\mathrm{d}^2u}{\mathrm{d}t^2} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}u}{\mathrm{d}\tau}\frac{1}{T}\right) = \frac{\mathrm{d}}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}t}\left(\frac{\mathrm{d}u}{\mathrm{d}\tau}\frac{1}{T}\right) = \frac{\mathrm{d}^2u}{\mathrm{d}\tau^2}\frac{1}{T^2} \tag{A-17}$$

scale time:

$$\frac{\mathrm{d}^2u}{\mathrm{d}\tau^2}\frac{1}{T^2} + ku + au^2 + bu^3 + 2\zeta\omega_n\frac{\mathrm{d}u}{\mathrm{d}\tau}\frac{1}{T} = f\sin(\omega\tau T) \tag{A-18}$$

scale only space:

$$\ddot{\tilde{u}}h + k\tilde{u}h + a(\tilde{u}h)^2 + b(\tilde{u}h)^3 + 2\zeta\omega_n\dot{\tilde{u}}h = f\sin(\omega t) \tag{A-19}$$

combine both:

$$\frac{\mathrm{d}^2\tilde{u}}{\mathrm{d}\tau^2}\frac{h}{T^2} + k\tilde{u}h + (\tilde{u}h)^2 + b(\tilde{u}h)^3 + 2\zeta\omega_n\frac{\mathrm{d}\tilde{u}}{\mathrm{d}\tau}\frac{h}{T} = f\sin(\omega\tau T) \tag{A-20}$$

mass normalise:

$$\frac{\mathrm{d}^2\tilde{u}}{\mathrm{d}\tau^2} + T^2\left(k\tilde{u} + ah\tilde{u}^2 + bh^2\tilde{u}^3\right) + 2\zeta\omega_n\frac{\mathrm{d}\tilde{u}}{\mathrm{d}\tau}T = f\frac{T^2}{h}\sin(\omega\tau T) \tag{A-21}$$

$$\tilde{u}'' + \tilde{k}\tilde{u} + \tilde{a}\tilde{u}^2 + \tilde{b}\tilde{u}^3 + 2\zeta\tilde{\omega}_n\tilde{u}' = \tilde{f}\sin(\tilde{\omega}\tau) \tag{A-22}$$

$$\frac{\tilde{k}}{k} = T^2, \qquad \frac{\tilde{a}}{a} = hT^2, \qquad \frac{\tilde{b}}{b} = h^2T^2, \qquad \frac{\tilde{\omega}_n}{\omega_n} = T, \qquad \frac{\tilde{f}}{f} = \frac{T^2}{h}, \qquad \frac{\tilde{\omega}}{\omega} = T \tag{A-23}$$

For good numerical performance, we impose $\tilde{k} = 1$ and $\max\{\tilde{a}, \tilde{b}\} = 1$. Therefore,

$$T = \frac{1}{\sqrt{k}}, \qquad h_a = \frac{1}{aT^2} = \frac{k}{a}, \qquad h_b = \frac{1}{T\sqrt{b}} = \sqrt{\frac{k}{b}} \tag{A-24}$$

$$h = \min\{h_a, h_b\} = \min\left\{\frac{k}{a}, \sqrt{\frac{k}{b}}\right\} \tag{A-25}$$

A-2-2 Eigenvector scaling - nonlinear stiffness

Mass-normalised quantities are denoted with subscript m, quantities corresponding to max-1 displacement eigenvectors are denoted with subscript u. The modal transformation for one eigenmode is defined as $\mathbf{x} = \boldsymbol{\phi} q$. The ratio between the eigenvectors is defined as:

$$\boldsymbol{\phi}_{\mathrm{m}} = \boldsymbol{\phi}_{\mathrm{u}} \alpha \tag{A-26}$$

Each scaling of the eigenmode should not influence the contribution of the mode to the total displacement:

$$\mathbf{x} = \boldsymbol{\phi}_{\mathrm{m}} q_{\mathrm{m}} = \boldsymbol{\phi}_{u} q_{u} \quad \longrightarrow \quad \boldsymbol{\phi}_{\mathrm{u}} \alpha q_{m} = \boldsymbol{\phi}_{\mathrm{u}} q_{u} \tag{A-27}$$

Yielding for each eigenmode i:

$$q_{\mathrm{m}_i} = \frac{q_{\mathrm{u}_i}}{\alpha_i} \tag{A-28}$$

Consider γ as the nonlinear modal force term, depending on the modal displacements: $\gamma(q_1, q_2, ..., q_L)$ with $q_1, q_2, ..., q_L$ the modal displacements of all modes included in the ROM.

For one mode i, the equation of motion writes:

$$\boldsymbol{\phi}_{\mathrm{m}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{m}_i} \ddot{q}_{\mathrm{m}_i} + \boldsymbol{\phi}_{\mathrm{m}_i}^T \mathbf{K} \boldsymbol{\phi}_{\mathrm{m}_i} q_{\mathrm{m}_i} + \gamma(q_{\mathrm{m}_1}, q_{\mathrm{m}_2}, ...) = \boldsymbol{\phi}_{\mathrm{m}_i}^T \mathbf{F} \tag{A-29}$$

replace the mass-normalised eigenvectors and modal displacements:

$$\alpha_i \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{u}_i} \alpha_i \ddot{q}_{\mathrm{u}_i} \frac{1}{\alpha_i} + \boldsymbol{\phi}_{\mathrm{u}_i}^T \alpha_i \mathbf{K} \boldsymbol{\phi}_{\mathrm{u}_i} \alpha_i q_{\mathrm{u}_i} \frac{1}{\alpha_i} + \gamma_{\mathrm{m}}(\frac{q_{\mathrm{m}_1}}{\alpha_1}, \frac{q_{\mathrm{m}_2}}{\alpha_2}, ...) = \boldsymbol{\phi}_{\mathrm{u}_i}^T \alpha_i \mathbf{F} \tag{A-30}$$

$$\alpha_i \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{u}_i} \ddot{q}_{\mathrm{u}_i} + \alpha_i \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{K} \boldsymbol{\phi}_{\mathrm{u}_i} q_{\mathrm{u}_i} + \gamma_{\mathrm{m}}(\frac{q_{\mathrm{m}_1}}{\alpha_1}, \frac{q_{\mathrm{m}_2}}{\alpha_2}, ...) = \boldsymbol{\phi}_{\mathrm{u}_i}^T \alpha_i \mathbf{F} \tag{A-31}$$

$$\boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{u}_i} \ddot{q}_{\mathrm{u}_i} + \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{K} \boldsymbol{\phi}_{\mathrm{u}_i} q_{\mathrm{u}_i} + \frac{1}{\alpha_i} \gamma_{\mathrm{m}}(\frac{q_{\mathrm{m}_1}}{\alpha_1}, \frac{q_{\mathrm{m}_2}}{\alpha_2}, ...) = \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{F} \tag{A-32}$$

$$\boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{M} \boldsymbol{\phi}_{\mathrm{u}_i} \ddot{q}_{\mathrm{u}_i} + \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{K} \boldsymbol{\phi}_{\mathrm{u}_i} q_{\mathrm{u}_i} + \gamma_{\mathrm{u}_i}(q_{\mathrm{u}_1}, q_{\mathrm{u}_2}, ...) = \boldsymbol{\phi}_{\mathrm{u}_i}^T \mathbf{F} \tag{A-33}$$

Here, γ is rewritten such that the α terms are implemented in the coefficients of all nonlinear terms. For example, if γ contains one cubic stiffness term and the first mode is considered:

$$\gamma_{\mathrm{m}_{122}}^{(1)} = b_{\mathrm{m}_{122}}^{(1)} q_{\mathrm{m}_1} q_{\mathrm{m}_2}^2 \tag{A-34}$$

$$\gamma_{\mathrm{u}_{122}}^{(1)} = \frac{1}{\alpha_1} b_{\mathrm{m}_{122}}^{(1)} \frac{q_{\mathrm{u}_1}}{\alpha_1} \left(\frac{q_{\mathrm{u}_2}}{\alpha_2}\right)^2 = \frac{b_{\mathrm{m}_{122}}^{(1)}}{\alpha_1^2 \alpha_2^2} q_{\mathrm{u}_1} q_{\mathrm{u}_2}^2 = b_{\mathrm{u}_{122}}^{(1)} q_{\mathrm{u}_1} q_{\mathrm{u}_2}^2 \tag{A-35}$$

So for each nonlinear stiffness term,

$$b^{(r)}_{\mathrm{u}_{ijk}} = \frac{b^{(r)}_{\mathrm{m}_{ijk}}}{\alpha_r \alpha_i \alpha_j \alpha_k}, \qquad a^{(r)}_{\mathrm{u}_{ij}} = \frac{a^{(r)}_{\mathrm{m}_{ij}}}{\alpha_r \alpha_i \alpha_j} \tag{A-36}$$

The max-1 based modal linear quantities can be directly obtained from multiplication with the max-1 based eigenvectors.

A-2-3 Nonlinear material damping

For linear systems, it is common to define linear damping, originating from energy losses in the material. In the hysteresic model, the damping is related to the stress inside the material. For harmonic motion ($u(t) = u_0 \cos \omega t$), the stress is defined as:

$$\sigma = E' \varepsilon + \frac{E''}{\omega} \frac{\mathrm{d}\varepsilon}{\mathrm{d}t} \tag{A-37}$$

which is known as the Kelvin-Voigt model. The storage term (related to stiffness) and the dissipation term are proportional to each other (for fixed ω). This implies that every stiffness force should have a companion damping force. For a linear stiffness term, the companion damping force writes:

$$F_{\mathrm{d}} = \frac{h}{\omega} \dot{u} \tag{A-38}$$

where u is displacement, ω is the frequency of excitation and h is the damper constant, which can be related to the Young's moduli. For linear underdamped systems, the highest amplitude of vibration can be found at it's resonant frequency, which is therefore usually substituted Eq. (A-38) for ω.

For each nonlinear stiffness term, the companion damping term can be calculated, resulting in a system obeying the material behaviour stated in Eq. (A-37).

If the strain equation contains the term $\varepsilon = \varepsilon_0 u^3$ resulting in cubic stiffness like $F = ku^3$, then the strain-rate and stress write:

$$\frac{\mathrm{d}\varepsilon}{\mathrm{d}t} = \frac{\partial \varepsilon}{\partial u} \frac{\partial u}{\partial t} = 3\varepsilon_0 u^2 \dot{u} \tag{A-39}$$

$$\sigma = E' \varepsilon + \frac{E''}{\omega} \frac{\mathrm{d}\varepsilon}{\mathrm{d}t} = E' \varepsilon_0 u^3 + 3 \frac{E''}{\omega} \varepsilon_0 u^2 \dot{u} \tag{A-40}$$

This yields a nonlinear damping force, also known as the Van der Pol term:

$$F_{\mathrm{d-NL}} = \frac{h}{\omega} u^2 \dot{u} \tag{A-41}$$

Appendix B

Supplementary material for Chapter 3

B-1 Additional material

Peak value

The forced Duffing equation including viscous damping writes:

$$\ddot{x} + \delta\dot{x} + \alpha x + \beta x^3 = \gamma\cos(\omega t) \tag{B-1}$$

Using the method of harmonic balancing, the solutions can be found to obey the following equation [27]:

$$\left[\left(\omega^2 - \alpha - \frac{3}{4}\beta z^2\right) + (\delta\omega)^2\right] z^2 = \gamma^2 \tag{B-2}$$

or

$$\frac{z^2}{\gamma^2} = \frac{1}{\left[\left(\omega^2 - \alpha - \frac{3}{4}\beta z^2\right) + (\delta\omega)^2\right]} \tag{B-3}$$

However, we are only looking for the limit point, where $\frac{z}{\gamma}$ is maximum. At this point, it can be assumed that the first term in the denominator is close to zero, yielding:

$$\frac{z_{\max}}{\gamma} \approx \frac{1}{\delta\omega} \tag{B-4}$$

which reveals that indeed, for increasing forcing (and thus eigenfrequency), the compliance decreases at the limit point. Form Eq. (B-3), for different forcing levels the limit point can be calculated numerically, which is shown in Figure B-1. Here, indeed the inverse proportionality to the vibration frequency is visible, as predicted in Eq. (B-4).

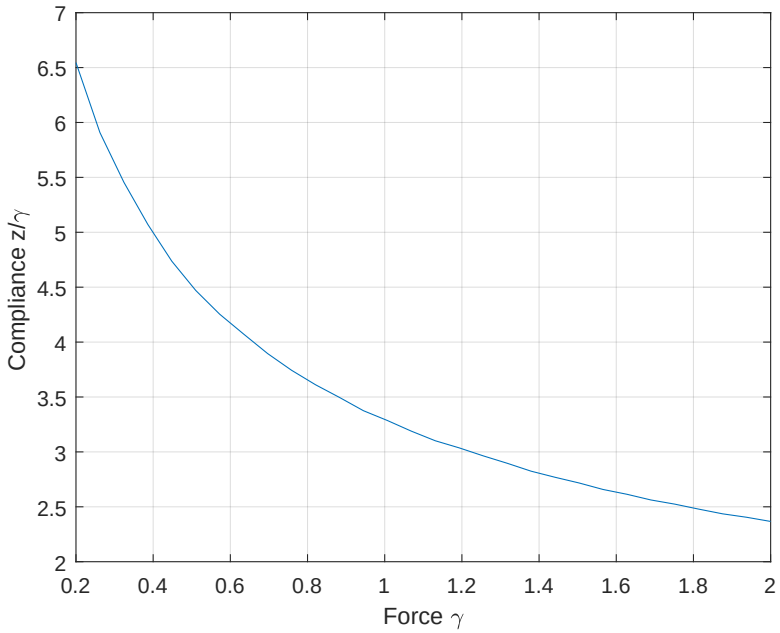


Figure B-1: Maximum compliance of the duffing oscillator as a function of forcing level.

B-1-1 Square membrane including bending stiffness

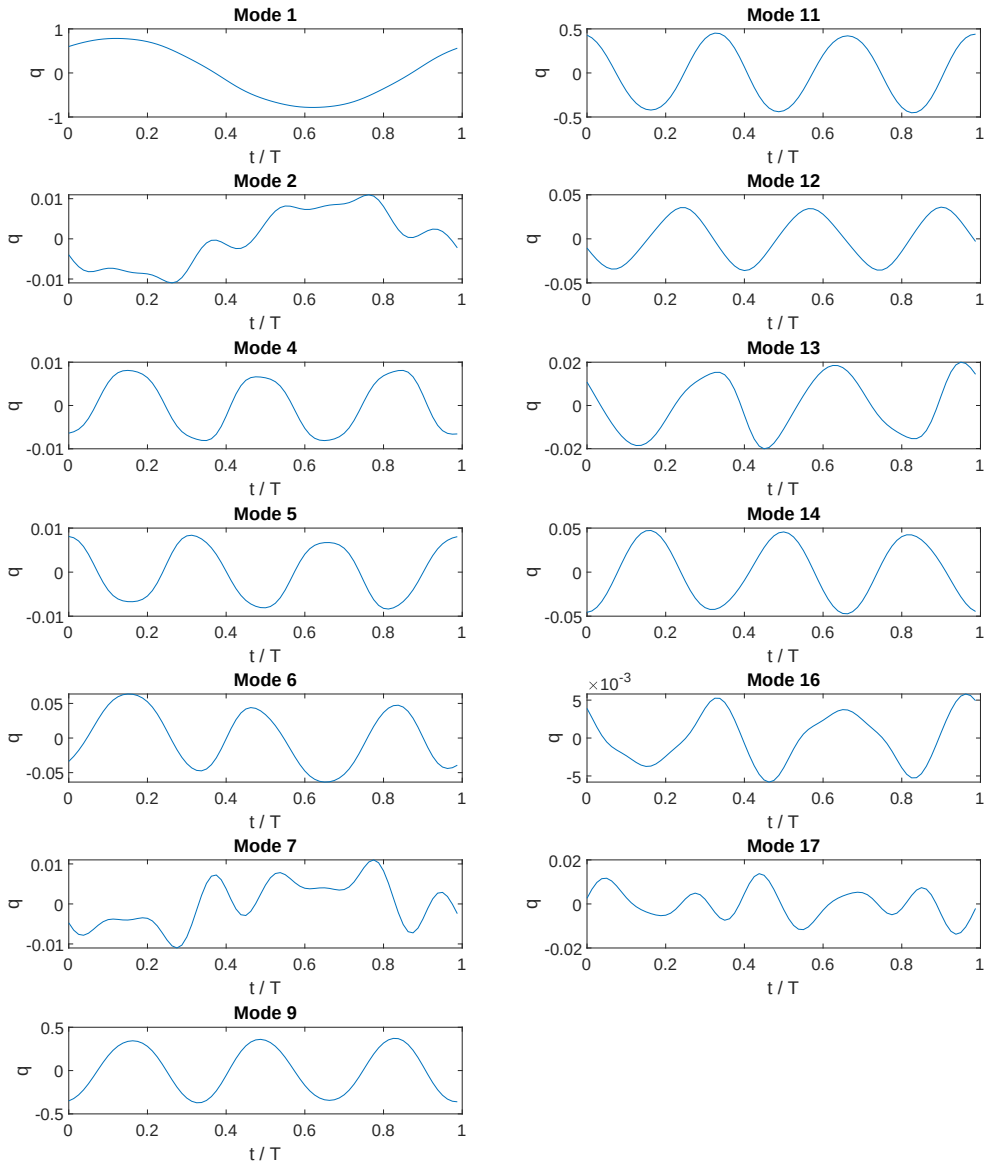


Figure B-2: Modal time signal of stable solution along the resonance curve at $\omega_f = 1.2\omega_1$ (the red dot in Figure 3-25a)

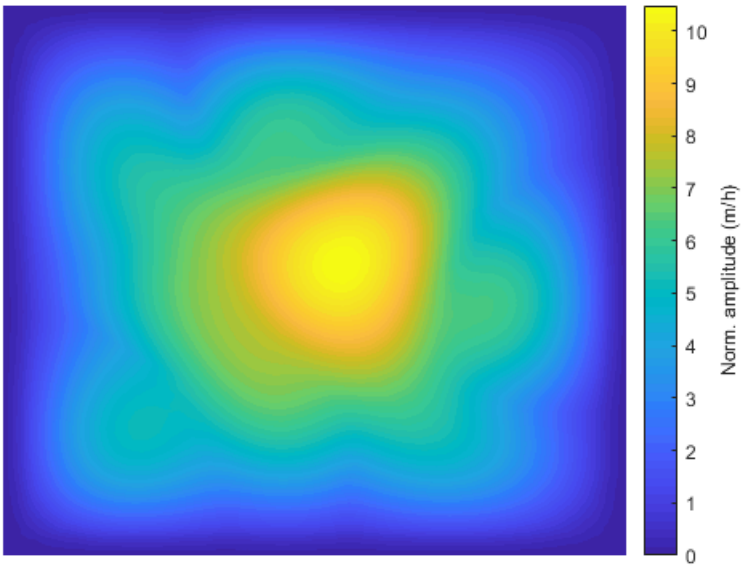


Figure B-3: Vibration amplitude at $\omega_{\mathrm{f}} = 1.2\omega_1$ (the red dot in Figure 3-25a)

B-1-2 Shooting - internal resonance

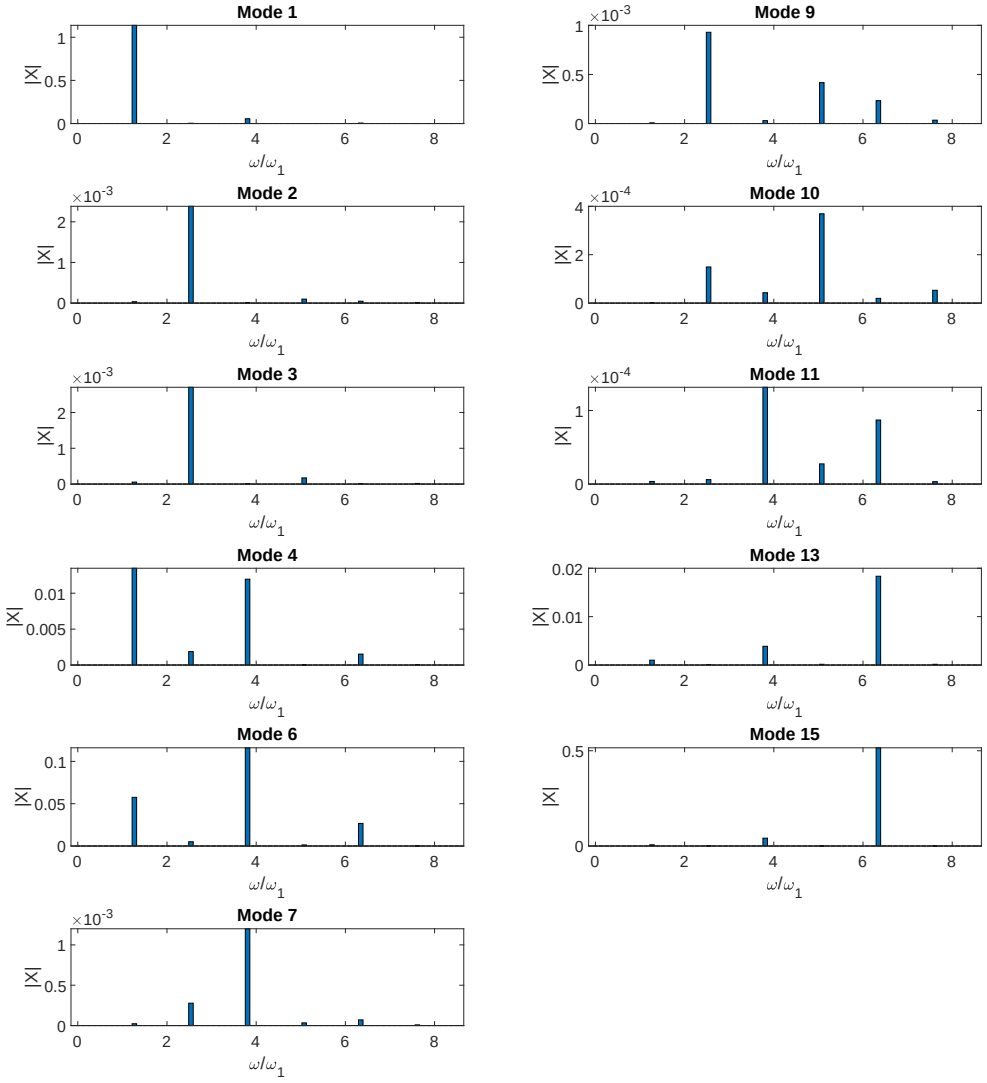


Figure B-4: FFT of modal signal of stable solution along the resonance curve at $\omega_{\mathrm{f}} = 2.53\omega_1$ (the red dot in Figure 3-42)

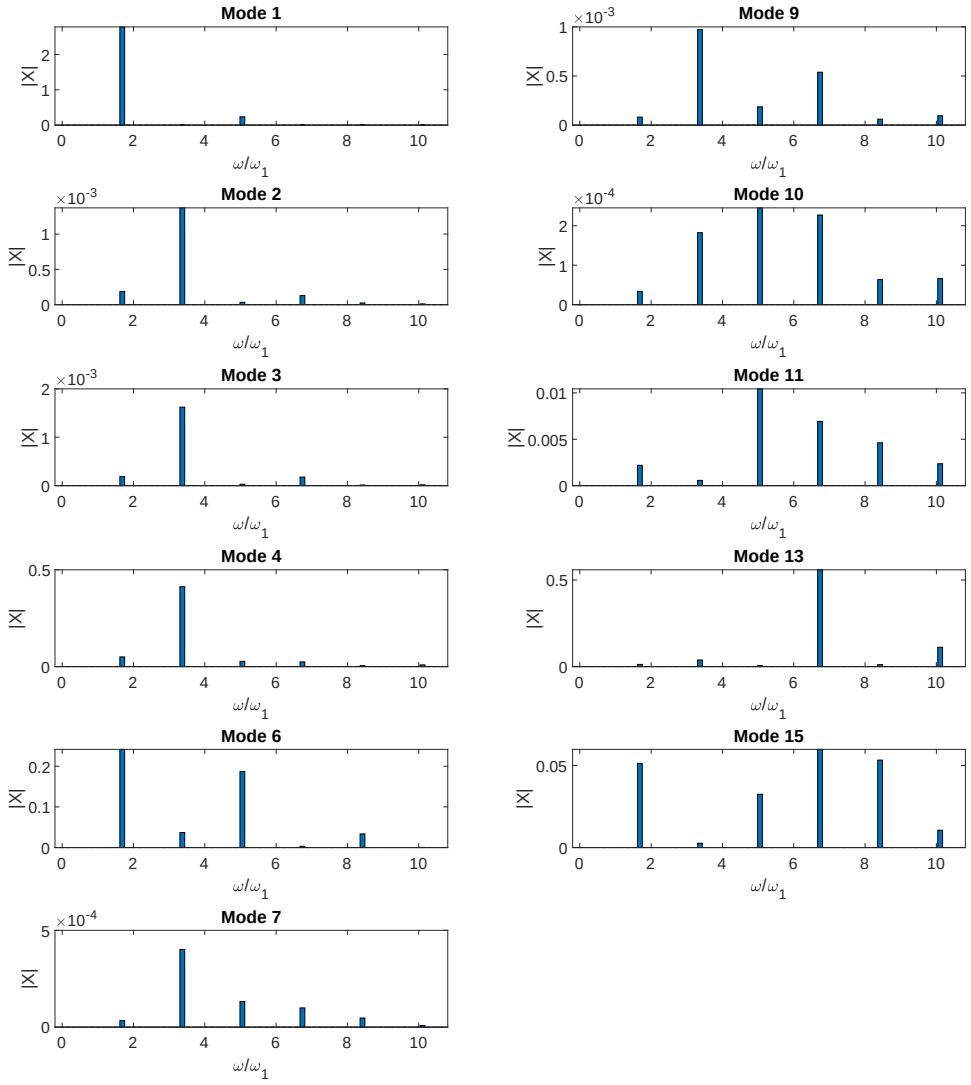


Figure B-5: FFT of modal signal of stable solution along the resonance curve at $\omega_{\mathrm{f}} = 3.37\omega_1$ (the black dot in Figure 3-42)

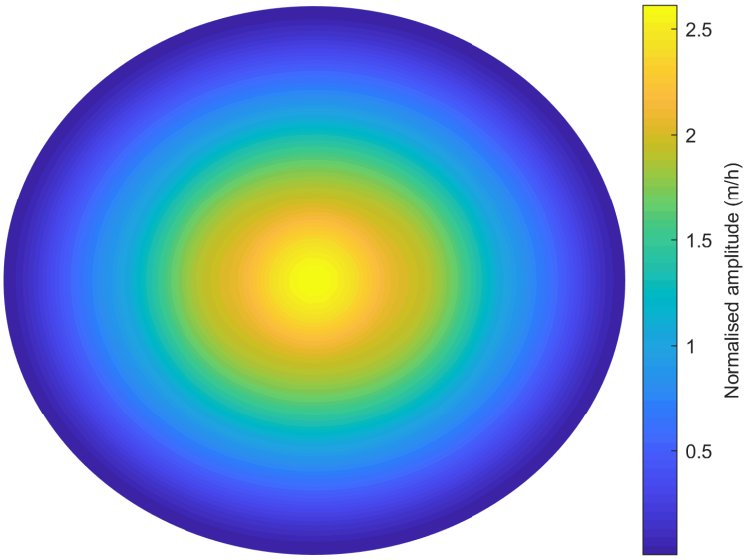


Figure B-6: Vibration amplitude at $\omega_f = 2.53\omega_1$ (the red dot in Figure 3-42)

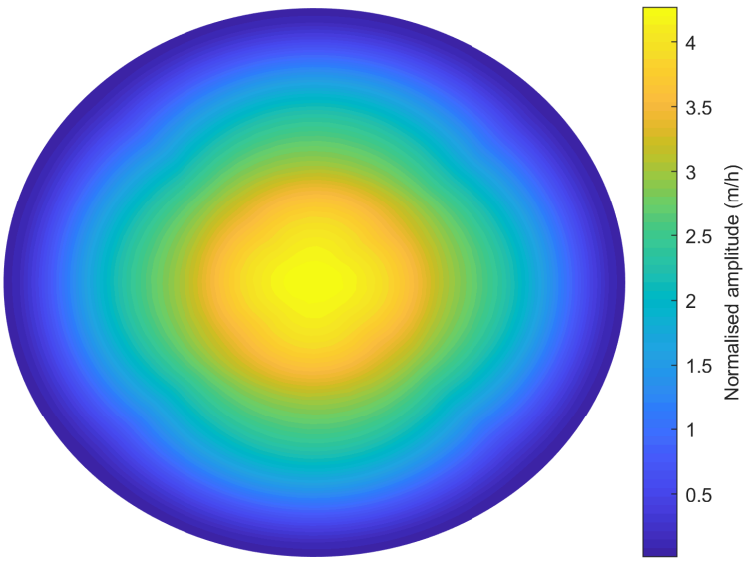


Figure B-7: Vibration amplitude at $\omega_f = 3.37\omega_1$ (the black dot in Figure 3-42)

B-1-3 Coefficient of variance

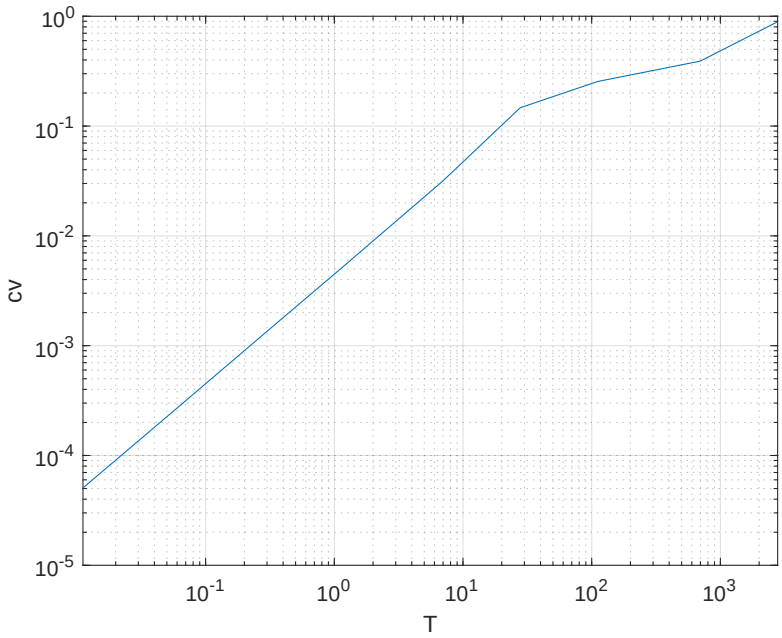


Figure B-8: Coefficient of variance as a function of noise temperature for a membrane.

Appendix C

Supplementary material for Chapter 4

C-1 Supporting material

C-1-1 Numerical accuracy

In this part, the accuracy of the numerical ODE solver is investigated. First the model used is introduced. Thereafter, the method of validation is explained followed by results.

Model

The model used is the same as presented in section 4-3, although in this case only the first 5 unique eigenmodes (listed in Table C-2) are included in the ROM.

Table C-1: Properties circular graphene membrane

Parameter	Value	Description
E	1014.9 GPa	Youngs modulus graphene
ν	0.16	Poisson's ratio
T_0	$0.0340\,\mathrm{N\,m^{-1}}$	Pretension
h	0.335 nm	Membrane thickness
R	300 nm	Radius
ρ	$2209\,\mathrm{kg/m^3}$	Density

Numerical stability

To investigate the performance of the numerical ODE solver, the total energy in the system is monitored. This comprises both linear energy (kinetic and potential) and the complete nonlinear set (both storage and coupling energy, as explained in section C-2-1). The equations describing the nonlinear coupling energy increase rapidly with the number of included modes,

Table C-2: First 5 normalised unique eigenfrequencies of a circular membrane

Index	Waveindex(t,r)	ω_n/ω_1	Shape
1	(0,1)	1	axisymmetric
2	(1,1)	1.5934	
4	(2,1)	2.1358	
6	(0,2)	2.2959	axisymmetric
7	(3,1)	2.6539	

and become very expensive to evaluate for large systems. Therefore, only five modes are included in the ROM.

It is assumed that for larger systems, the global error remains approximately the same (for the same solver settings), as the internal error handling of the solver remains the same.

Simulations

Table C-3: Simulation settings numerical stability. This ROM consist only of the first 5 unique modes.

Parameter	Value	Description
n_{modes}	5	Number of modes in the reduced order model
ζ	0.5e-10	Viscous damping active on all modes
n_r	40	Number of simulations per parameter set
n_{per}	20	Number of periods simulated
χ	0.65	Fraction of initial energy in fundamental mode
$\tilde{V}_0$	4	Relative initial velocity of the first mode

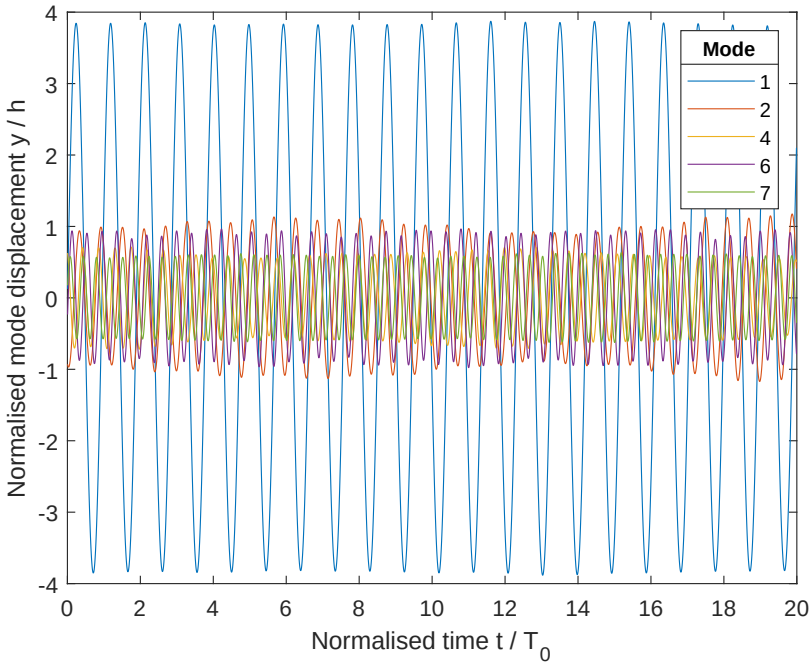


Figure C-1: Modal displacement

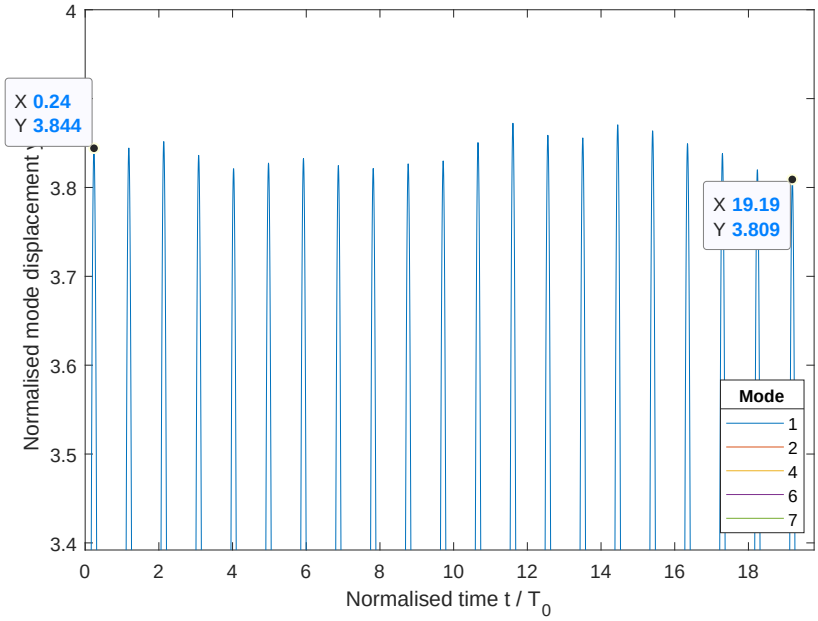


Figure C-2: Modal displacement - vertical zoom

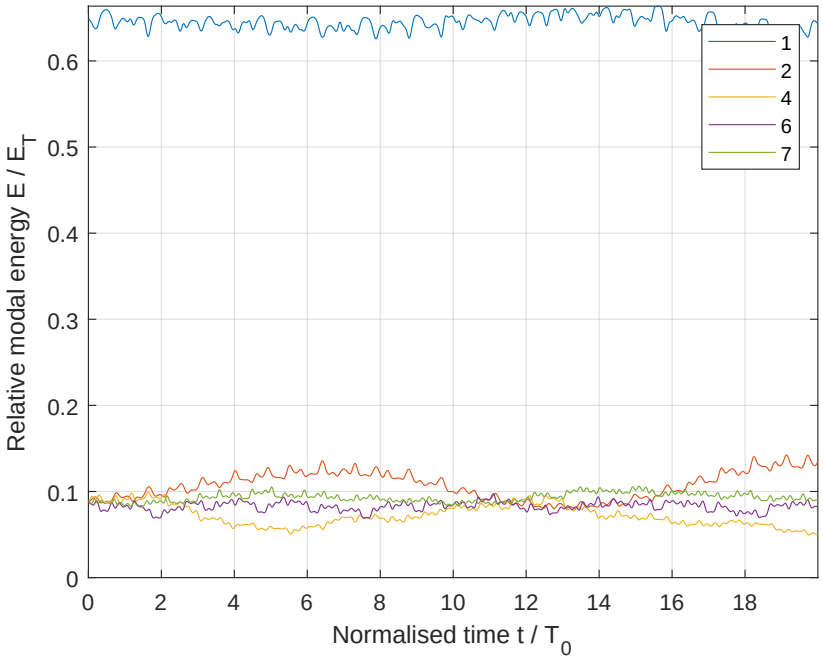


Figure C-3: Modal energy

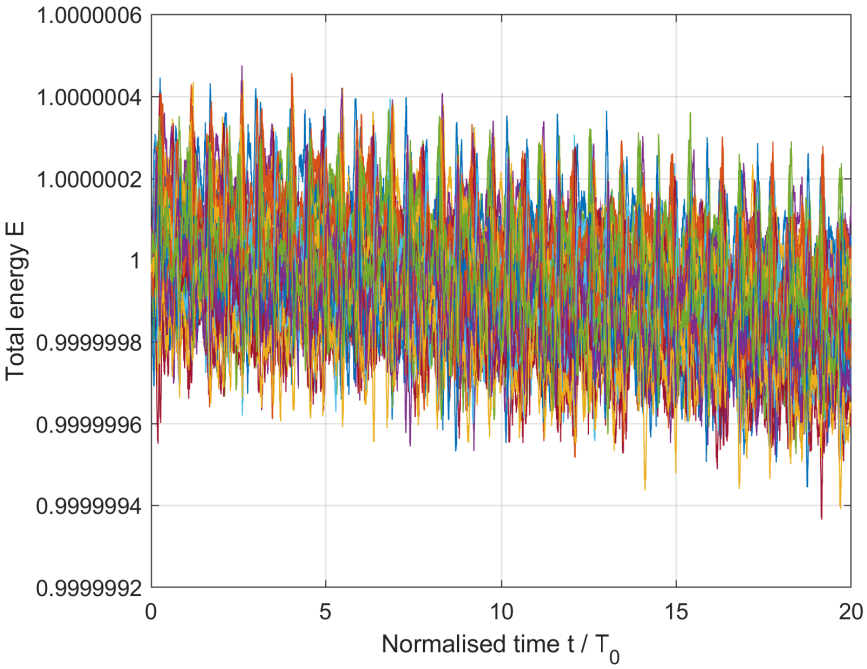


Figure C-4: Total energy

C-1 shows the modal response for one simulation. In Figure C-2, the amplitude of the

fundamental mode is shown more detailed, revealing the slight decay. This may also be observed in Figure C-3, in which the modal energy is shown. Note that, because also the coupling energy is included here, it looks a little bumpy. However, summing all modal energies yields the total system energy, which is almost constant. In Figure C-4 the total energy is shown for all simulations. Of coarse, a very small energy drift is present, due to finite numerical precision. In the next section, the consequence of this energy drift is investigated.

Numerical damping

Energy loss due to numerical effects can be expressed in an equivalent loss factor η_{num}, in order to compare it significance:

$$\eta_{\text{num}} = \frac{\Delta U}{2\pi U} = \frac{\Delta U_{\text{tot}}}{2\pi U n} = \frac{\Delta U_{\text{tot}}}{U t \omega} \tag{C-1}$$

Here, U_{tot} indicates the total energy lost during the whole simulation, and ω equals the mean radial frequency over the whole timespan. The total number of periods is estimated using:

$$n = \frac{t}{T} = \frac{t\omega}{2\pi} \tag{C-2}$$

Note that there is also very little viscous damping present, of which the effect is also included in η_{num}. Last, the Absolute Tolerance mainly influences the solver accuracy. Simulation results are listed for three different values of ζ in Tab C-4, C-5 and C-6.

Table C-4: Simulation results concerning numerical stability. Reported values correspond to the average of all simulations. Note that for a linear system, $\eta = 2\zeta$

Parameter	Value	Description
ζ	5e-10	Viscous damping active on all modes
η_{coup}	2.8602e-05	Mode coupling damping
η_{num}	1.3681e-09	Numerical damping
AbsTol	1e-12	Absolute Tolerance ODE solver
n_{pnt}	200	Number of time steps per fundamental period

Table C-5: Simulation results concerning numerical stability. Reported values correspond to the average of all simulations. Note that for a linear system, $\eta = 2\zeta$

Parameter	Value	Description
ζ	5e-11	Viscous damping active on all modes
η_{coup}	2.8601e-05	Mode coupling damping
η_{num}	1.8945e-10	Numerical damping

From the results, it becomes clear that the absolute tolerance effectively sets the accuracy of the ODE solver. A intuitive guess for the numerical damping could be:

$$\eta_{\text{num}} = \frac{n_{\text{pnt}} \text{AbsTol}}{2\pi y_{\text{rep}}} \tag{C-3}$$

Table C-6: Simulation results concerning numerical stability. Reported values correspond to the average of all simulations. Note that for a linear system, $\eta = 2\zeta$

Parameter	Value	Description
ζ	5e-16	Viscous damping active on all modes
η_{coup}	2.8601e-05	Mode coupling damping
η_{num}	4.7118e-11	Numerical damping

which should actually be squared for energy equivalence.

Conclusion

Concluding from the results, it becomes clear that the numerical error is effectively set by the Absolute Tolerance of the ODE solver, resulting in a numerical loss factor of about n_{pnt}AbsTol. For the used settings, the numerical damping remains more than 5 orders of magnitude lower than the damping through mode coupling.

C-1-2 Equivalent damping ratio

For a single dof oscillator including viscous damping, a damping ratio ζ, loss factor η and Q-factor can be determined. As the loss factor indicates the relative amount of energy lost with respect to the stored energy, this value can be useful to compare different damping mechanisms.

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{C-4}$$
$$\ddot{x} + 2\zeta\omega_n\dot{x} + \omega_n^2 x = 0 \tag{C-5}$$

$$\zeta = \frac{c}{2\omega_n m}, \quad \eta = \frac{\Delta U}{2\pi U} = 2\zeta, \quad Q = \frac{1}{\eta} \tag{C-6}$$

To compare the different terms, they are written in the same form as the viscous damping term: $F \propto c\dot{q}_1$. The coefficient can be a function of amplitude now, and velocity and displacement amplitude are related by assuming simple harmonic motion:

$$q = \sin(\omega t) \tag{C-7}$$
$$\dot{q} = \omega \sin(\omega t) \tag{C-8}$$
$$\rightarrow q \propto \frac{1}{\omega}\dot{q} \tag{C-9}$$

Table C-7: Damping models for harmonic motion $x = A\cos(\omega_d t)$ and unit mass $m = 1$, found from [28]

Type	Equation of motion	η
Viscous	$\ddot{x} + c\dot{x} + \omega_n^2 x = 0$	$2\zeta\frac{\omega_d}{\omega_n}$
Fluid	$\ddot{x} + c\lvert\dot{x}\rvert\dot{x} + \omega_n^2 x = 0$	$\frac{3}{4\pi}c\frac{\omega_d}{\omega_n}A$
Displ.-based	$\ddot{x} + cx^2\mathrm{sgn}(\dot{x}) + \omega_n^2 x = 0$	$\frac{4}{\pi}c\frac{1}{\omega_d^2}A$

C-2 Mathematical derivations

C-2-1 Elastic potential

The linear energy at each time instant writes:

$$E_{\mathrm{L}} = \frac{1}{2}k^{(1)}q_1^2 + \frac{1}{2}k^{(2)}q_2^2 + \frac{1}{2}m^{(1)}\dot{q}_1^2 + \frac{1}{2}m^{(2)}\dot{q}_2^2 \tag{C-10}$$

Here, the kinetic and potential terms interchange the energy. When taking for $\hat{q}$ the amplitude of vibration, the linear energy stored in mode 1 can be written as:

$$E_{\mathrm{L}}^{(1)} = \frac{1}{2}k^{(1)}\hat{q}_1^2 \tag{C-11}$$

The nonlinear elastic potential of each spring can be found by integrating the force:

$$E_{112}^{(1)} = \int F du = \int b_{112}^{(1)}q_1^2 q_2 dq_1 = \frac{1}{3}b_{112}^{(1)}q_1^3 q_2 \tag{C-12}$$

$$E_{111}^{(2)} = \int F du = \int b_{111}^{(2)}q_1^3 dq_2 = b_{111}^{(2)}q_1^3 q_2 \tag{C-13}$$

$$E_{1112} = c\ f(q_1^3 q_2) = E_{112}^{(1)} = E_{111}^{(2)} \tag{C-14}$$

Therefore, also the coefficients should be related to each other:

$$\frac{1}{3}b_{112}^{(1)} = b_{111}^{(2)} \tag{C-15}$$

In the same way other relations can be constructed.

The total nonlinear potential writes:

$$E_{\mathrm{NL}} = \frac{1}{4}b_{111}^{(1)}q_1^4 + \frac{1}{3}b_{112}^{(1)}q_1^3 q_2 + \frac{1}{2}b_{122}^{(1)}q_1^2 q_2^2 + b_{222}^{(1)}q_1 q_2^3 + \frac{1}{4}b_{222}^{(2)}q_2^4 \tag{C-16}$$

$$E_{\mathrm{NL}} = E_{1111}(q_1^4) + E_{1112}(q_1^3, q_2) + E_{1122}(q_1^2, q_2^2) + E_{1222}(q_1, q_2^3) + E_{2222}(q_2^4) \tag{C-17}$$

Here, the energies with mixed indices allow for coupling, the others only for storage.

$$E_{\text{nl}} = E_{\text{storage}} + E_{\text{coupling}} = E_{\text{nls}} + E_{\text{nlc}} \tag{C-18}$$

However, the energy stored solely in the first mode is just the first term:

$$E_{\text{NL}}^{(1)} = \frac{1}{4} b_{111}^{(1)} q_1^4 \tag{C-19}$$

C-2-2 Relative error nonlinear stiffness coefficients

By checking Eq. (C-15) after determining the stiffness coefficients using the STEP method, the accuracy of both coefficients can be determined. Using all equality relations, the accuracy of the complete analysis can be determined. The relative error between two coefficients b_1 and b_2 that should be equal to each other could be calculated as:

$$\epsilon = \frac{b_1 - b_2}{(b_1 + b_2)/2} \tag{C-20}$$

To prevent division by zero, the cubic stiffness of the fundamental modes is taken as a representative value (absolute values are considered):

$$b_{\text{rep}} = \max\{b_{111}^{(1)}, (b_1 + b_2)/2\} \tag{C-21}$$

yielding

$$\epsilon = \frac{b_1 - b_2}{b_{\text{rep}}} \tag{C-22}$$

C-2-3 Energy transfer between modes

The coupling terms in the right hand side of the equations can be seen as nonlinear springs, connecting both modes. Each mode can add or subtract energy to the spring, and in this way energy can go from one mode to another. If we assume harmonic motion for both modes, the work done on the nonlinear spring can be estimated

$$\begin{aligned} F &= b_{112}^1 q_1^2 q_2 && \text{(C-23)} \\ q_1 &= \hat{q}_1 \sin(\omega_1 t) && \text{(C-24)} \\ q_2 &= \hat{q}_2 \sin(\omega_2 t) && \text{(C-25)} \\ \dot{q}_1 &= \hat{q}_1 \omega_1 \cos(\omega_1 t) && \text{(C-26)} \\ n\omega_1 &= m\omega_2 && \text{(C-27)} \end{aligned}$$

The energy transmitted from mode 1 to the $b_{112}^{(1)}$ spring (during m periods of mode 1) can be found as

$$P = Fv = b^1_{112} q_1^2 q_2 \dot{q}_1 \tag{C-28}$$

$$W = \int_0^{T_1} P dt \tag{C-29}$$

$$W^{(1)}_{112} = b^1_{112} \hat{q}_1^3 \hat{q}_2 \frac{1}{8} \left(\frac{\cos(2\pi m\delta)}{\delta} - \frac{\cos(2\pi m\delta_3)}{\delta_3} - \frac{\cos(2\pi m\sigma)}{\sigma} + \frac{\cos(2\pi m\sigma_3)}{\sigma_3} \right) \tag{C-30}$$

$$\delta = \frac{\omega_2 - \omega_1}{\omega_1},\ \delta_3 = \frac{\omega_2 - 3\omega_1}{\omega_1}, \sigma = \frac{\omega_1 + \omega_2}{\omega_1}, \sigma_3 = \frac{3\omega_1 + \omega_2}{\omega_1}\ ,\ T_1 = m\frac{2\pi}{\omega_1} \tag{C-31}$$

The expression can be split up in an energy and a frequency dependent part:

$$W^{(1)}_{112} = c_1 E_{1112} f(\omega_1, \omega_2) \tag{C-32}$$

C-2-4 Power loss free vibration

$$P = \frac{\Delta U}{\Delta t} = \frac{U_{i+1} - U_i}{t_{i+1} - t_i} \tag{C-33}$$

Here the stored energy U is determined form the sum of the elastic and kinetic energy. The elastic energy also includes storage terms from the nonlinear elastic potential.

C-2-5 Instant frequency

Because the system is nonlinear, the free-vibration frequency changes with amplitude of vibration. Therefore, instead of just assuming the linear eigenfrequency, the instantaneous frequency is taken, which is defined as

$$\omega_{inst}(t) = \frac{\mathrm{d}\phi(t)}{\mathrm{d}t} \tag{C-34}$$

For a simple cosine, this can easily be verified:

$$s(t) = A\cos(\omega t + \theta) = A\cos(\phi(t)) \tag{C-35}$$

The instantaneous frequency can also be seen as the geometric centre of the Amplitude Spectral Density (ASD), which is calculated as:

$$f_{\text{inst}} = \frac{\int_0^\infty f X(f) df}{\int_0^\infty X(f) df} \tag{C-36}$$

Here, $X(f)$ is determined from the Fast Fourier Transform of the time signal.

C-2-6 Bending stiffness

Note that bending stiffness of a circular membrane is also dependent on the radius (using Kirchhoff-Love Plate theory for circular plates):

$$k_{bend} \propto \frac{Eh^3}{R^2} \tag{C-37}$$

The total linear stiffness is the sum of both. However, for thin structures, bending stiffness can be ignored:

$$\frac{k_{\text{pre}}}{k_{\text{bend}}} = \sigma h \frac{R^2}{Eh^3} = \varepsilon_{pre} Eh \frac{R^2}{Eh^3} = \varepsilon_{pre} \frac{R^2}{h^2} \tag{C-38}$$

Appendix D

2dof example

This chapter describes the interaction between in-plane modes and out-of-plane modes, as introduced in section 2-4. A very simple, 2dof model is introduced, resembling a string or simplified membrane. For this model, the different approaches to handle in-plane modes (as shown in section 2-4-1 and 2-4-2) are applied. However, because the model contains only 2 degrees of freedom, the nonlinear stiffness can also be determined exact. Thereafter, the analytically derived exact differential equations are compared to the equations yielding from the approximation methods by calculating the frequency response.

D-1 The simplified model

Here we consider a simplified model for a nonlinear membrane shown in figure D-1. The model is intended to capture the essential physics of nonlinear membrane motion in only a 2 degree of freedom system. Two masses are connected to each other and the walls by three identical strings, each string having a stiffness area product $\kappa = EA$, where E is the Young's modulus and A the cross sectional area. Only motion that is symmetric upon reflection in the central dashed line along the y-axis, such that the central string always has to stay horizontal. With this constraint we focus on the fundamental membrane modes only. Therefore the motion can be considered fully by addressing the motion of left mass only (then the position of the other mass is obtained by mirror symmetry).

D-1-1 Initial condition

In the initial state the membrane is flat, with all three strings having a length L_0 towards the center of the masses. In addition there is initial tension applied on the strings by extending them over a distance $L_t = L_0 - L_i$, with their actual starting length without any tension being $L_i = L_0 - L_t$, with $L_t \ll L_0$. As a consequence the initial tension in the string is $F_t = \kappa L_t / L_i$. The energy stored in the string is therefore $E_s = \frac{1}{2}\frac{\kappa}{L_i}(L - L_i)^2$, for simplicity we define $k = \frac{\kappa}{L_i}$.

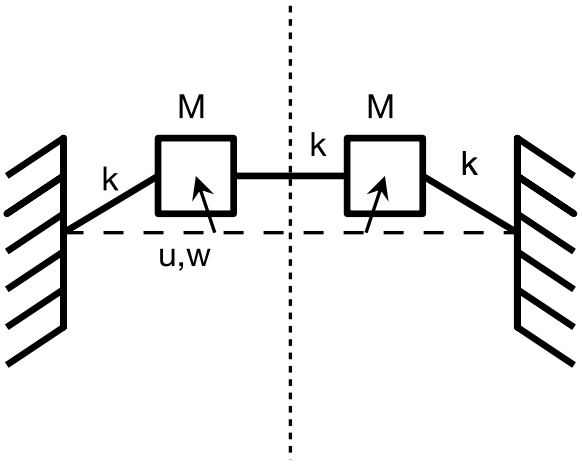


Figure D-1: Simplified model for a nonlinear membrane.

D-1-2 Energy expressions

Now the mass M is displaced by a distance u in the x-direction and a distance w in the y-direction. For this configuration the length of the left string becomes:

$$L_l = \sqrt{(L_0 + u)^2 + w^2} \tag{D-1}$$

The elongation of the middle spring equals two times the horizontal displacement of the left mass, because the displacement of the other mass is in opposite direction. Therefore, a displacement of the left mass u leads to a length change $2u$. The length of the middle string equals:

$$L_m = L_0 - 2u \tag{D-2}$$

Now the spring energy stored in the left string is:

$$E_l = \frac{1}{2}k(L_l - L_i)^2 \tag{D-3}$$

Because we consider only the left half of the system, we on only take half the energy of the middle spring into account, giving in addition a factor 1/2.

$$E_m = \frac{1}{2}\frac{1}{2}k(L_m - L_i)^2 \tag{D-4}$$

And the kinetic energy is;

$$E_{kin} = \frac{1}{2}M(\dot{u}^2 + \dot{w}^2) \tag{D-5}$$

D-1-3 Equations of motion

We obtain for the Lagrangian:

$$\mathcal{L} = E_{kin} - (E_l + E_m) = \frac{M}{2}(\dot{u}^2 + \dot{w}^2) - \frac{1}{2}k(L_l - L_i)^2 - \frac{1}{4}k(L_m - L_i)^2) \tag{D-6}$$

From this the equations of motion can be derived:

$$M\ddot{w} + k(1 - \frac{L_i}{L_l})w = F_y \tag{D-7}$$

$$M\ddot{u} + k[3u + L_i(1 - \frac{L_0 + u}{L_l})] = F_x \tag{D-8}$$

Linearization gives (using $L_l \approx L_0 + u$):

$$M\ddot{u} + 3ku = F_x \tag{D-9}$$

$$M\ddot{w} + kL_t\frac{w}{L_0} = F_y \tag{D-10}$$

When we linearise the equations of motion, they decouple. The equation for horizontal displacement describes just a simple spring-mass-spring system, where the right spring has double stiffness as it's midpoint is virtually fixed. The second equation mimics a string with a slider at one end, which seems plausible since in our system, the middle spring remains horizontal. Here, kL_t mimics the pretension, and $\frac{w}{L_0}$ approximates the tangents, yielding the vertical component of the pretension force.

In more mathematical terms, the linear solutions of the equations of motion are thus just the sinusoidal motions of the mass along the x-axis $(u_0 e^{i\omega_2 t}, 0)$ and along the y-axis $(0, w_0 e^{i\omega_1 t})$ with $\omega_2^2 = \frac{3k}{M}$ and $\omega_1^2 = \frac{kL_t}{ML_0}$. Since $L_t \ll L_0$, it follows that $\omega_1 \ll \omega_2$. The corresponding linear modeshapes $\phi(u, w)$ are $\phi_1 = (0, 1)$ and $\phi_2 = (1, 0)$.

D-2 Method Muravyov and Rizzi

According to the paper we impose a static displacement $\mathbf{X}_c = +\phi_1 q_1 = (u = 0, w = q_1)$ on the structure. The static force can be obtained from the equations of motion, by setting the accelerations equal to zero. For the linear case, this results in a force vector:

$$\mathbf{F}_L(\phi_1 q_1) = (0, k\frac{L_t}{L_0}q_1) \tag{D-11}$$

For the nonlinear case this results in (see appendix):

$$\mathbf{F}_T(+\phi_1 q_1) \approx (\frac{1}{2}k\frac{L_i}{L_0^2}q_1^2 - \frac{3}{8}k\frac{L_i}{L_0^4}q_1^4 + O(q_1^6), k\frac{L_t}{L_0}q_1 + \frac{1}{2}k\frac{L_i}{L_0^3}q_1^3 - \frac{3}{8}k\frac{L_i}{L_0^5}q_1^5 + O(q_1^7)) \tag{D-12}$$

Note that we are neglecting all terms higher than third order here. In principle higher orders could be kept, although this would significantly complicate matters. Since these higher order terms are (as far as I can judge) kept by finite element methods like Comsol, it might be that in the following the results of the coefficients could depend on the actual value chosen for q_1. It is therefore of importance to keep q_1 small enough, such that $(q_1/L_0)^2 \ll 1$. All analysis is only exactly valid in this limit.

$$\mathbf{F}_{NL1} = \mathbf{\Gamma}(+\phi_1 q_1) = \mathbf{F}_T(+\phi_1 q_1) - \mathbf{F}_L(+\phi_1 q_1) = (\frac{1}{2}k\frac{L_i}{L_0^2}q_1^2, \frac{1}{2}k\frac{L_i}{L_0^3}q_1^3) \tag{D-13}$$

$$\mathbf{F}_{NL2} = \mathbf{\Gamma}(-\phi_1 q_1) = (\frac{1}{2}k\frac{L_i}{L_0^2}q_1^2, -\frac{1}{2}k\frac{L_i}{L_0^3}q_1^3) \tag{D-14}$$

$$\tilde{\mathbf{F}}_{NL1} = \mathbf{\Phi}^T\mathbf{F}_{NL1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbf{F}_{NL1} = \begin{pmatrix} \frac{1}{2}k\frac{L_i}{L_0^3}q_1^3 \\ \frac{1}{2}k\frac{L_i}{L_0^2}q_1^2 \end{pmatrix} = [a_{11}^r]q_1^2 + [b_{111}^r]q_1^3 \tag{D-15}$$

$$\tilde{\mathbf{F}}_{NL2} = \mathbf{\Phi}^T\mathbf{F}_{NL2} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\mathbf{F}_{NL2} = \begin{pmatrix} -\frac{1}{2}k\frac{L_i}{L_0^3}q_1^3 \\ \frac{1}{2}k\frac{L_i}{L_0^2}q_1^2 \end{pmatrix} = [a_{11}^r]q_1^2 - [b_{111}^r]q_1^3 \tag{D-16}$$

Solving these 2 equations for 2 unknowns gives:

$$[a_{11}^r] = \begin{pmatrix} 0 \\ \frac{1}{2}k\frac{L_i}{L_0^2} \end{pmatrix} \tag{D-17}$$

$$[b_{111}^r] = \begin{pmatrix} \frac{1}{2}k\frac{L_i}{L_0^3} \\ 0 \end{pmatrix} \tag{D-18}$$

According similar methods all other elements of the a and b matrix of vectors can be constructed, as outlined in [1], this leads to a reduced equation of motion as shown in equations (3) and (5) of [1].

D-2-1 Other coefficients

The other coefficients can be determined in a similar way(see appendix). We find:

$$[a_{22}^r] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{D-19}$$

$$[b_{222}^r] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{D-20}$$

$$[a_{12}^r] = \begin{pmatrix} k\frac{L_i}{L_0^2} \\ 0 \end{pmatrix} \tag{D-21}$$

$$[b^r_{112}] = \begin{pmatrix} -k\frac{L_i}{L_0^3} \\ 0 \end{pmatrix} \tag{D-22}$$

$$[b^r_{122}] = \begin{pmatrix} -k\frac{L_i}{L_0^3} \\ 0 \end{pmatrix} \tag{D-23}$$

D-2-2 Equations of motion

Now the equations of motion of the approximate model (containing max cubic order terms) can be written:

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + a_{12}^{(1)}q_1q_2 + b_{111}^{(1)}q_1^3 + b_{122}^{(1)}q_1q_2^2 = F^y \tag{D-24}$$

$$m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 + b_{112}^{(2)}q_1^2q_2 = F^x \tag{D-25}$$

From this we can see that the in-plane eigenmode, has no source of nonlinearity from itself ($a_{22}^{(2)}$ and $b_{222}^{(2)}$ equal zero), it has only nonlinear coupling to the out-of-plane mode. The quadratic term in the equation for out-of-plane motion ($(a_{12}^{(1)})$ indicates softening behaviour, lowering the effective stiffness of this mode. If we consider first the static case with no external forcing on the second mode, we can write q_2 in terms of q_1 and substitute this back in the equation of motion for q_1, yielding the effective stiffness of this mode.

$$k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 + b_{112}^{(2)}q_1^2q_2 = 0$$

$$q_2 = \frac{-a_{11}^{(2)}q_1q_1}{k_1^{(2)} + b_{112}^{(2)}q_1^2}$$

Substitute $q_2(q_1)$ in the static equation (see appendix) for q_1 (and use $L_i \approx L_0$):

$$b_{111-eff}^{(1)} = b_{111}^{(1)} - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}} = \frac{1}{2}k\frac{L_i}{L_0^3} - \frac{1}{6}k\frac{L_i^2}{L_0^4}\frac{1}{3}k\frac{L_i}{L_0^3} \tag{D-26}$$

D-3 Farbod's method - Static condensation

According to the method applied by Farbod, when considering only out-of-plane modes, the coefficients of [1] are too stiff. For motion in the linear eigenmode ϕ_1 in the y-direction, the mass can also move in the x-direction to lower the elastic energy.

As can be seen in equation (18) a displacement according ϕ_1 results in a nonzero force in the x-direction, which might be compensated by a motion along the ϕ_2 mode (u displacement). Since the resonance frequency $\omega_2 \gg \omega_1$, this motion can occur virtually instantaneous when

the motion is at ω_1 and the inertia of the system for movement along the ϕ_2 mode can be neglected.

In that case, only a displacement $w = q_1$ along the y-direction is imposed, and the force in the x-direction should be zero to have equilibrium.

We return to equation (17):

$$\mathbf{F}_T = k[3u + L_i(1 - \frac{L_0 + u}{L_l})], k(1 - \frac{L_i}{L_l})w) \tag{D-27}$$

$$\begin{aligned} \mathbf{F}_T(+\phi_1 q_1 + \phi_2 q_2) &= (k[(L_0 - \frac{L_i}{\sqrt{(1 + \frac{q_2}{L_0})^2 + \frac{q_1^2}{L_0^2}}})(1 + \frac{q_2}{L_0}) - (L_0 - L_i - q_2)], \quad \text{(D-28)} \\ & \quad k(L_0 - \frac{L_i}{\sqrt{(1 + \frac{q_2}{L_0})^2 + \frac{q_1^2}{L_0^2}}})\frac{q_1}{L_0}) \quad \text{(D-29)} \end{aligned}$$

Now we look for the solution for which the forces along the x-direction are zero. The equation above is difficult to solve, but for small q_1/L_0 one can simply take the linear stiffness $2k$ in the x-direction and set it equal to the x-direction force generated by the out-of-plane displacement. So $3kq_2 = -\frac{kL_i}{2L_0^2}q_1^2$, (note that this result can also be obtained by the equation above, by setting the horizontal force equal to zero) gives:

$$q_{2,eq} = -\frac{1}{6}\frac{L_i}{L_0^2}q_1^2 \tag{D-30}$$

The motion in the fundamental mode will thus follow the parabolic curve $(u(q_1), w(q_1)) = (-\frac{1}{6}\frac{L_i}{L_0^2}q_1^2, q_1)$. In that case the nonlinear force becomes:

$$\begin{aligned} \mathbf{F}_T(+\phi_1 q_1 + \phi_2(-\frac{1}{6}\frac{L_i}{L_0^2}q_1^2)) &= (0, \\ & \quad k(L_0 - \frac{L_i}{\sqrt{(1 - \frac{1}{6}\frac{L_i}{L_0^3}q_1^2)^2 + \frac{q_1^2}{L_0^2}}})\frac{q_1}{L_0}) \end{aligned}$$

Now for small q_1/L_0 and $L_i \approx L_0$, which is true for small L_t, one obtains:

$$\begin{aligned} F_{Tw} &\approx k(L_0 - \frac{L_i}{\sqrt{(1 - \frac{1}{3}\frac{L_i}{L_0^3}q_1^2) + \frac{q_1^2}{L_0^2}}})\frac{q_1}{L_0} \approx kL_i(L_t/L_i + 1 - \frac{1}{\sqrt{1 + \frac{2}{3}\frac{q_1^2}{L_0^2}}})\frac{q_1}{L_0} \\ &\approx k\frac{L_t}{L_0}q_1 - \frac{1}{3}k\frac{L_i}{L_0^3}q_1^3 + \frac{1}{6}k\frac{L_i}{L_0^5}q_1^5 \end{aligned}$$

Working this further out according to the method of [1], results in the same cubic stiffness as the effective cubic stiffness obtained by the first method. However, when no in-plane modes are involved, the first method gives a nonlinear stiffness which is a factor 1.5 too high.

$$[a^r_{11}] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{D-31}$$

$$[b^r_{111}] = \begin{pmatrix} \frac{1}{3}k\frac{L_i}{L_0^3}q_1^3 \\ 0 \end{pmatrix} \tag{D-32}$$

D-4 Approximations

D-4-1 Nonlinear mass

The in-plane mode has a nonzero modal mass, so at some point this may influence the dynamic response of the out-of-plane mode. If we consider harmonic motion around the first eigenfrequency (of the out-of-plane mode), the in-plane mode will still follow it's forcing terms, as it's mass has small influence at $\omega = \omega_1 << \omega_2$. To investigate the lowest-order influence, we neglect the b_{112} and b_{122} stiffness, as they result in higher order coupling. So we obtain the following simplified equations of motion:

$$\begin{aligned} m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + a_{12}^{(1)}q_1q_2 + b_{111}^{(1)}q_1^3 &= \sin(\omega_f t) && \text{(D-33)} \\ m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 &= 0 && \text{(D-34)} \end{aligned}$$

Consider motion of the out-of-plane mode following the forcing term:

$$\begin{aligned} q_1 &= \hat{q_1}\sin(\omega_f t) \\ q_1^2 &= \hat{q_1}^2\sin(\omega_f t)^2 = \hat{q_1}^2\frac{1}{2}(1-\cos(2\omega_f t)) \end{aligned}$$

We can consider the quadratic coupling term in the in-plane equation of motion as an external forcing, yielding:

$$m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 = -a_{11}^{(2)}q_1q_1 = -a_{11}^{(2)}\hat{q_1}^2\frac{1}{2}(1-\cos(2\omega_f t))$$

The static force yields a static displacement:

$$q_{2-\text{static}} = -\frac{a_{11}^{(2)}}{2k_1^{(2)}}\hat{q_1}^2 = -\beta\frac{1}{2}\hat{q_1}^2 \tag{D-35}$$

The dynamic force can be analysed using Laplace (only looking at steady state solutions) (use $\omega_{2f} = 2\omega_f$):

$$m^{(2)}\ddot{q}_2 + k_1^{(2)} q_2 = a_{11}^{(2)} \hat{q}_1^2 \frac{1}{2}\cos(\omega_{2f} t) = f(\omega_{2f}) \tag{D-36}$$

$$Q_2(-m^{(2)}(\omega_{2f})^2 + k_1^{(2)}) = F(\omega_{2f}) \tag{D-37}$$

$$\frac{Q_2(\omega_{2f})}{F(\omega_{2f})} = \frac{1}{(k_1^{(2)} - m^{(2)}(\omega_{2f})^2} = \frac{1}{k_1^{(2)}} \frac{1}{1 - (\frac{\omega_{2f}}{\omega_2})^2} \tag{D-38}$$

$$q_{2-\text{dynamic}} = \frac{a_{11}^{(2)}}{2k_1^{(2)}} \hat{q}_1^2 \frac{1}{1 - (\frac{2\omega_f}{\omega_2})^2} \cos(2\omega_f t) = \frac{1}{2}\beta\alpha(\omega_f)\hat{q}_1^2 \cos(2\omega_f t) \tag{D-39}$$

The total solution then writes:

$$q_2 = q_{2-\text{static}} + q_{2-\text{dynamic}} = -\beta\frac{1}{2}\hat{q}_1^2 + \frac{1}{2}\beta\alpha(\omega_f)\hat{q}_1^2 \cos(2\omega_f t) \tag{D-40}$$

$$q_2 = -\beta\hat{q}_1^2 \frac{1}{2}(1 - \alpha\cos(2\omega_f t)) = -\beta\hat{q}_1^2 \left[\frac{1}{2}\alpha(1 - \cos(2\omega_f t)) - \frac{1}{2}(\alpha - 1)\right] \tag{D-41}$$

$$q_2 = -\beta\alpha\hat{q}_1^2 \frac{1}{2}\left[(1 - \cos(2\omega_f t)) - (1 - \frac{1}{\alpha})\right] \tag{D-42}$$

Recall

$$q_1^2 = \hat{q}_1^2 \sin(\omega_f t)^2 = \hat{q}_1^2 \frac{1}{2}(1 - \cos(2\omega_f t))$$

$$q_2 = -\beta\alpha\left[q_1^2 - \hat{q}_1^2 \frac{1}{2}(1 - \frac{1}{\alpha})\right] = -\beta\alpha\left[q_1^2 - \epsilon\right] \tag{D-43}$$

For $\omega_f = 0$, $\epsilon = 0$, so $q_2 \geq 0$. For $\omega_f < 2\omega_n$, $\alpha \geq 1$, so in this domain $0 < \epsilon < \frac{1}{2}\hat{q}_1^2$. Actually, ϵ is translating the squared sine downward. This causes q_2 to also take negative values, meaning that the in-plane mode is also moving towards the centre.

This can be inserted in the out-of-plane equation of motion:

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \left(-\beta\alpha\left[q_1^2 - \epsilon\right]\right) + b_{111}^{(1)} q_1^3 = \sin(\omega_f t) \tag{D-44}$$

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \epsilon - \beta\alpha a_{12}^{(1)} q_1^3 + b_{111}^{(1)} q_1^3 = \sin(\omega_f t) \tag{D-45}$$

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \epsilon + q_1^3 (b_{111}^{(1)} - \beta\alpha a_{12}^{(1)}) = \sin(\omega_f t) \tag{D-46}$$

As ϵ depends on $\hat{q}_1$ which is not analythic available, ϵ is removed. However, it is assumed to lower the maximum value of q_2. As q_2 seems to have a softening effect on q_1, this softening can be seen as an upper bound, so probably the amplitude of q_1 will be slightly lower. When putting back all the original terms in the equation,

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + q_1^3 \left(b_{111}^{(1)} - \frac{a_{11}^{(2)} a_{12}^{(1)}}{k_1^{(2)}} \frac{1}{1 - (\frac{2\omega_f}{\omega_2})^2}\right) = \sin(\omega_f t) \tag{D-47}$$

Here, we find again the same factor we found for the static analysis $\frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}$. However, an additional term governing the in-plane inertia is now visible. This term becomes infinite at half the in-plane eigenfrequency, removing the cubic hardening effect of the out-of-plane mode itself. Therefore, the backbone curve will not pass this point.

Small in-plane damping (eg $Q > 100$) can be added, as long as the phase shift is small enough to neglect.

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + q_1^3\left(b_{111}^{(1)} - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}\frac{1}{\sqrt{(1-(\frac{2\omega_f}{\omega_2})^2)^2 + (2\zeta\frac{2\omega_f}{\omega_2})^2}}\right) = \sin(\omega_f t) \quad \text{(D-48)}$$

Damping can reduce the softening influence of the inertia, and can prevent the out-of-plane to lock when the in-plane mode starts to resonate. The above derivation is not exact, but it contains the features regarding strong mode interaction and reveals the dependencies.

One interesting extension could be to include in the in-plane equation the cubic coupling term $b_{112}^{(2)}q_1^2q_2$. This yields a linear stiffness which is parametrically excited, resulting in resonances at integer fractions like $2\omega_f = \frac{1}{n}\omega_2, n = 2, 3, 4...$ (see paper par res).

From this we can see that around the out-of-plane eigenfrequency, the in-plane inertia does not play a big role. However, at higher frequency, the dynamic stiffness of the in-plane motion reduces, by which the negative cubic correction term increases. At $\omega_f = \frac{1}{2}\omega_2$, the in-plane mode should resonate and the out-of-plane mode could get negative cubic stiffness creating instability.

To compare this with other ways of modeling nonlinear inertia, we approximate the fraction with a Taylor series, which is valid for small $\frac{\omega_f}{\omega_2}$ (see appendix):

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + b_{111-eff}^{(1)}q_1^3\left[1 - \frac{1}{2}\left(\frac{2\omega_f}{\omega_2}\right)^2\right] = \sin(\omega_f t) \quad \text{(D-49)}$$

The nonlinear intertia appears to scale with $q_1^3\omega^2$. Terms like $x\dot{x}^2$ or $\ddot{x}x^2$ (which could be introduced to the out-of-plane equation of motion), have the same scaling.

D-4-2 Nonlinear damping

The non-linear mass approximation incorporates the softening due to in-plane mass. If the in-plane mode has some viscous damping, it dissipates part of it's stored energy. However, energy can also flow from the out-of-plane mode to the in-plane mode, where it is dissipated. To account for this, an additional nonlinear damping term could be added to the NLM approximation. The dissipated energy during one cyce due to viscous damping writes:

$$\Delta U = 2\pi\hat{x}^2\omega_n\omega\zeta \quad \text{(D-50)}$$

The approximation term should dissipate the same amount of energy:

$$\Delta U_{1a} = 2\pi \hat{q_1}^2 \omega_n^{(1)} \omega_1 \zeta_{1a} = 2\pi \hat{q_2}^2 \omega_n^{(2)} \omega_2 \zeta_2 \tag{D-51}$$

$$\zeta_{1a} = \zeta_2 \left(\frac{\hat{q_2}}{\hat{q_1}}\right)^2 \frac{\omega_n^{(2)}}{\omega_n^{(1)}} \frac{\omega_2}{\omega_1} \tag{D-52}$$

Using the amplitude relation obtained before (only considering the harmonic part):

$$\hat{q_2} \approx \beta\alpha \hat{q_1}^2 \tag{D-53}$$

Substitute $\frac{\hat{q_2}}{\hat{q_1}} \approx \beta\alpha\hat{q_1}$:

$$\zeta_{1a} = \zeta_2 (\beta\alpha\hat{q_1})^2 \frac{\omega_n^{(2)}}{\omega_n^{(1)}} \frac{\omega_2}{\omega_1} \tag{D-54}$$

Note that, in this case, $\omega_2 \approx 2\omega_1$, $\beta \approx 1$, $1 < \alpha < \infty$, $\frac{\omega_n^{(2)}}{\omega_n^{(1)}} \approx 20$, so the effective damping ratio of this nonlinear damping term is at least 20 times higher than it's linear damping ratio. The added term to the equation of motion of the out-of-plane mode could be:

$$2\omega_n^{(1)} \zeta_{1a} \dot{q_1} \approx c q_1^2 \dot{q_1} \tag{D-55}$$

This would hold for excitation frequencies far away from the resonances of the in-plane mode, where $\alpha \approx 1$. Anyway, this leads to the conclusion that if structures suffer from amplitude or frequency depended damping, this could be caused by coupled-damping and can be described by a quadratic dependence of the viscous damping value.

D-5 Real string

To compare the validity of the proposed model introduced in the first section, it is compared here to the nonlinear model yielding from a real string. To do so, Rizzi's method is implemented in Comsol to obtain the coefficients. We define the structure to be a A_4 string, tuned to 440 Hertz. The string has diameter h of 1mm, a pre-stress of 1520 MPa, a total length of 50 cm and a density of 7850 kg/m^3.

Table D-1: Coefficients of a Comsol model and simplified model (Rizzi's method)

Parameter	Value	Description
ρ	7850 kgm^-3	density
σ	1520 MPa	pre-stress
D	1 mm	Diameter
L_{full}	0.5 m	Full length
E	210 GPa	Young's modulus

$$L_0 = \frac{1}{3} L_{full} \tag{D-56}$$

$$\epsilon = \frac{\sigma}{E}; \tag{D-57}$$

$$L_t = \epsilon L_0 \tag{D-58}$$

$$L_i = L_0 - L_t \tag{D-59}$$

Table D-2: Coefficients of a Comsol model and simplified model (Rizzi's method)

Coefficient	Simple model	Comsol
$k^{(1)}$	$7.2151 \cdot 10^3$	$1.1781 \cdot 10^4$
$k^{(2)}$	$2.9905 \cdot 10^6$	$6.5563 \cdot 10^6$
$m^{(1)}$	0.0015	0.0015
$m^{(2)}$	0.0015	0.0015
f_1	344.3421	440.0529
f_2	7010.3	10383
f_2/f_1	20.3586	23.596
$a_{11}^{(1)}$	0	0
$a_{11}^{(2)}$	$2.9688 \cdot 10^6$	$0.5113 \cdot 10^7$
$a_{12}^{(1)}$	$5.9376 \cdot 10^6$	$1.0225 \cdot 10^7$
$a_{12}^{(2)}$	0	0
$a_{22}^{(1)}$	0	0
$a_{22}^{(2)}$	0	0
$b_{111}^{(1)}$	$1.7813 \cdot 10^7$	$2.410 \cdot 10^7$
$b_{111}^{(2)}$	0	0
$b_{112}^{(1)}$	0	0
$b_{112}^{(2)}$	$-3.5626 \cdot 10^7$	$6.424 \cdot 10^7$
$b_{122}^{(1)}$	$-3.5626 \cdot 10^7$	$6.424 \cdot 10^7$
$b_{122}^{(2)}$	0	0
$b_{222}^{(1)}$	0	0
$b_{222}^{(2)}$	0	$3.9286 \cdot 10^8$

From the table, the following can be said about the resemblance:

- The modal masses are equal. This is expected as they arise just from the structure geometry

- The out-of-plane modal linear stiffness is about 30% less than the one yielding from FEM. The 2dof in-plane stiffness is about 50%, which could be due to the fact that the in-plane mode is actually the second in plane mode (when sorted from low to high stiffness) of the complete system. Usually, the approximation becomes less accurate for hgher modes.
- The coefficients of the nonlinear coupling stiffness terms are almost all in the same order of magnitude.
- The cubic coupling terms from the 2dof model have negative sign, where those from FEM have positive ones. Those negative stiffnesses are assumed to cause in-plane instability (see figure ...), as the FEM equations does not show this instability.
- The FEM analysis yields cubic nonlinear stiffness for the in-plane mode, which is not in the 2dof model.

D-5-1 Nonlinear in-plane stiffness

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_j} + \frac{\partial u_k}{\partial x_i}\frac{\partial u_k}{\partial x_j}\right) \tag{D-60}$$

$$\varepsilon_{xx} = \frac{\partial u_x}{\partial x_x} + \frac{1}{2}\left(\frac{\partial u_x}{\partial x_x}\frac{\partial u_x}{\partial x_x} + \frac{\partial u_y}{\partial x_x}\frac{\partial u_y}{\partial x_x} + \frac{\partial u_z}{\partial x_x}\frac{\partial u_z}{\partial x_x}\right) \tag{D-61}$$

For $\nu = 0$ and if only load in x-direction are considred, the resulting displacements are only in the direction of the load, leaving only one nonlinear term:

$$\varepsilon_{xx} = \frac{\partial u_x}{\partial x_x} + \frac{1}{2}\left(\frac{\partial u_x}{\partial x_x}\right)^2 \tag{D-62}$$

For a bar, loaded axially at the endpoint (endpoint displacement q), (stress and strain are constant over the volume), with unit volume, the elastic energy writes:

$$U = \frac{1}{2}\int \sigma\varepsilon \mathrm{d}V = \frac{1}{2}\int \varepsilon E\varepsilon \mathrm{d}V = \frac{1}{2}\varepsilon E\varepsilon \tag{D-63}$$

using

$$\sigma = \varepsilon E \tag{D-64}$$

$$u_x = xq \tag{D-65}$$

$$\frac{\partial u_x}{\partial x_x} = q \tag{D-66}$$

$$\varepsilon_{xx} = q + \frac{1}{2}q^2 \tag{D-67}$$

$$U = E(\frac{1}{2}q + \frac{1}{2}q^2)(q + \frac{1}{2}q^2) = \frac{1}{2}E(q^2 + \frac{1}{3}q^3 + \frac{1}{4}q^4) \tag{D-68}$$

The energy term which is dependent on q^4 originates from a cubic stiffness, which is also seen in the table.

However, here also a quadratic stiffness term arises, which is not present in the table. This term has the property that, for negative displacement, the energy becomes negative. The in-plane eigenmode is symmetric around the midpoint. This means that the 'quadratic stiffness energy' stored in the left part is the same as in the right part, but with opposite sign. Therefore, the contributions cancel out, and the effective quadratic stiffness equals zero.

To obtain the stiffness terms for the in-plane mode for the string, the displacement field of the mode can be inserted into the strain equation.

$$u_x = \hat{q} \sin x \tag{D-69}$$

$$\frac{\partial u_x}{\partial x_x} = \hat{q} \cos x \tag{D-70}$$

$$\varepsilon_{xx} = \hat{q} \cos x + \frac{1}{2}\hat{q}^2 \cos^2 x \tag{D-71}$$

$$U = \frac{1}{2}\int \sigma\varepsilon \mathrm{d}V = \frac{1}{2}\int \varepsilon E \varepsilon \mathrm{d}V = \frac{1}{2}AE \int \varepsilon_{xx}^2 \mathrm{d}x = \tag{D-72}$$

$$U = \frac{1}{2}AE \int_0^{2\pi} \left(\hat{q} \cos x + \frac{1}{2}\hat{q}^2 \cos^2 x\right)^2 \mathrm{d}x \tag{D-73}$$

$$U = \frac{1}{2}AE \int_0^{2\pi} \hat{q}^2 \cos^2 x + \hat{q}^3 \cos^3 x + \hat{q}^4 \frac{1}{4} \cos^4 x \mathrm{d}x \tag{D-74}$$

Recall

$$\int_0^{2\pi} \cos^3 x \mathrm{d}x = 0 \tag{D-75}$$

Therefore, the 'quadratic stiffness' energy term vanishes, so indeed the effective quadratic stiffness equals zero. As in the table, only the linear and cubic stiffness remain.

D-6 Numerical solution

It would be of interest to solve this system in AUTO or Matcont and see if the result of the differential equations (11) and (12) indeed can be well captured by both methods, and also see at what amplitude corrections are needed.

D-6-1 Statics

To obtain insight the nature of the differences, we consider first the static case. As load case, we put an vertical external force on the mass, so only the out-of-plane mode is directly excited. Then we solve for the out-of-plane and in-plane displacements, for each load step.

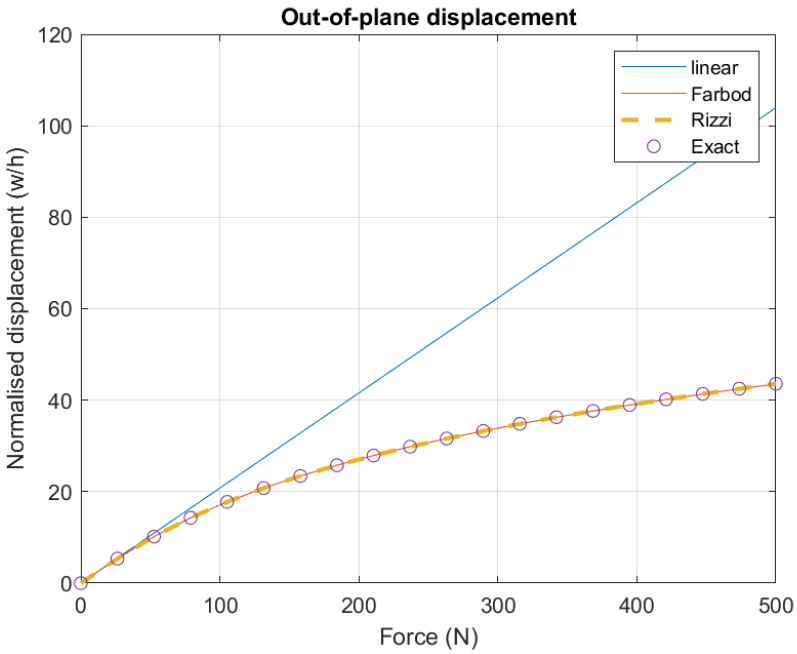


Figure D-2: Force-displacement diagram.

The result can be seen in figure D-2, where the resulting out-of-plane displacement is plotted versus the external force. The linear response seems reliable up to about 10 times the thickness. At higher displacements the response is clearly nonlinear. Both Farbod's and Rizzi's method seem to be able to describe the out-of-plane displacement correctly, as they match the exact solution perfectly.

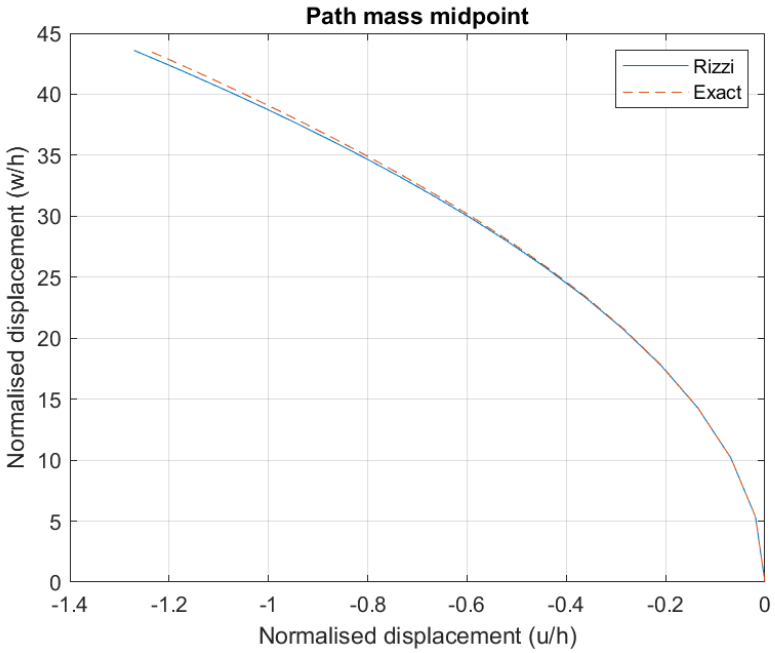


Figure D-3: Path mass mindpoint.

Figure D-3 shows the path of the mass midpoint, with respect to the undeformed configuration. As the linear solution and Farbod's method do not (directly) solve for the in-plane motion, they are not shown here. Rizzi's method seems to be able to describe the in-plane movement quite well, as the curves match quite good. Only at large deformations, the in-plane displacement yielding from Rizzi's method seems to deviate very slightly from the exact solution. Note that when using Farbod's method, using the relation between q_1 and q_2, the in-plane displacement could be calculated afterwards.

In conclusion, we can say that when only considering statics, both Farbod's and Rizzi's method can describe the nonlinear effects quite well. Computationally, Rizzi's method is more expensive as the number of equations to solve doubles, but it also directly yields the in-plane displacements.

D-6-2 Dynamics

To analyse the dynamic response of the structure, we use AUTO. The structure is subjected to a harmonic load, to obtain the frequency response. To prevent infinite resonance peaks, some artificial damping ζ is added to the equations of motion, ($\zeta = 0.002$).

Linear regime

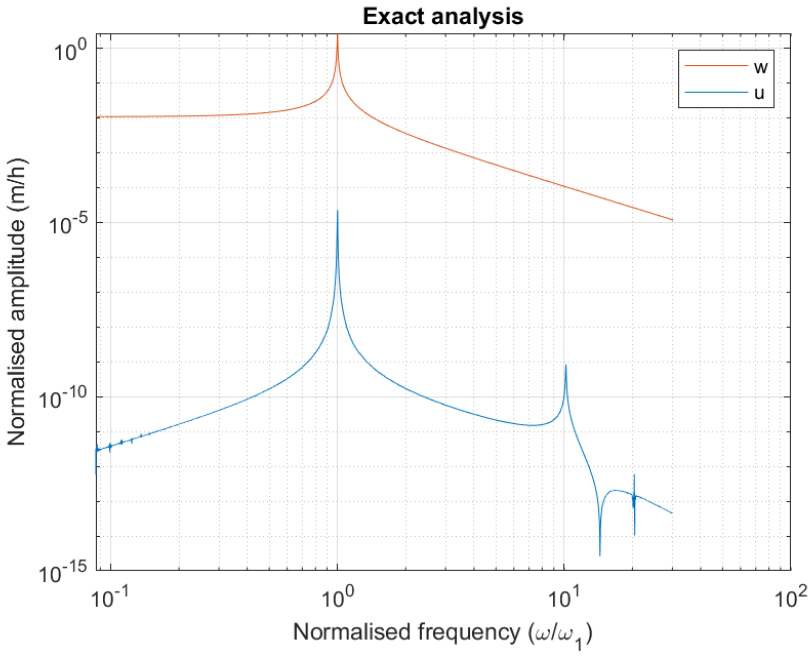


Figure D-4: Frequency response at low force level. Amplitude is normalised to string diameter. $F = 0.05\text{N}$

Considering very low excitation forces, all methods give the same result. As visible in figure D-4, the out-of-plane response looks quite linear. The in-plane response shows it's own resonance at $10\omega_1$. The actual resonance frequency equals $20\omega_1$, but because in-plane motion has two periods during one out-of-plane period, the resonance is found at half the true resonance frequency. From now on, we look only around the first resonance, as at higher frequency nothing special happens when increasing the excitation force.

Entering the nonlinear regime

When increasing the excitation force to 0.5N, hardening occurs, but still all methods match the exact solution. The responses are shown in figures D-5 and D-6

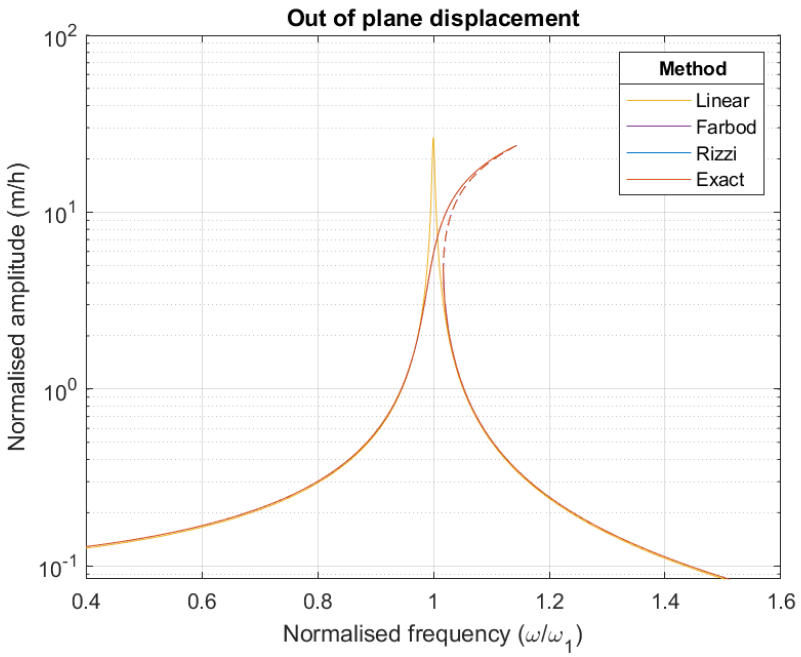


Figure D-5: Frequency response, entering the nonlinear regime. $F = 0.5\text{N}$

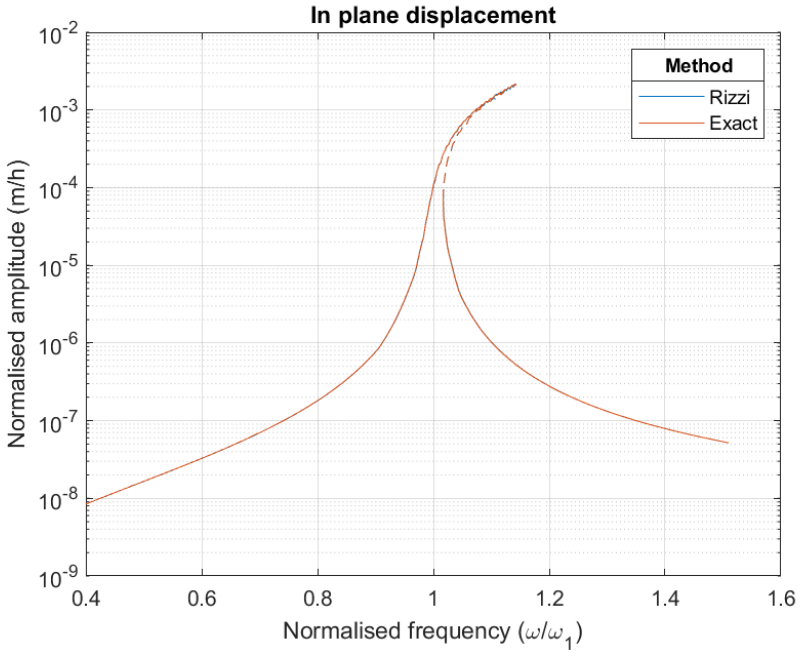


Figure D-6: Frequency response, entering the nonlinear regime. $F = 0.5\text{N}$

Nonlinear regime

After the excitation force is increased to 5N, small differences are visible (see figure D-5 and D-6). Farbod's method seems to be slightly stiffer than the exact solution, while Rizzi is still matching it. This is probably due to the softening effect of the in-plane mass, as found in equation D-49. Furthermore, Farbod's solution shows a resonance peak that has it's maximum value at higher frequency than the exact solution, as visible in figure D-8

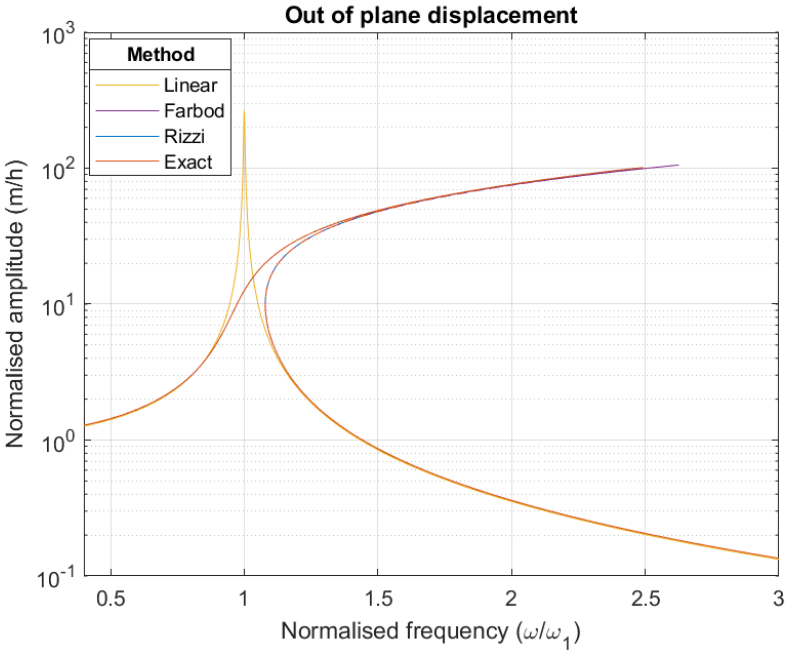


Figure D-7: Frequency response, further into the nonlinear regime. $F = 5\text{N}$

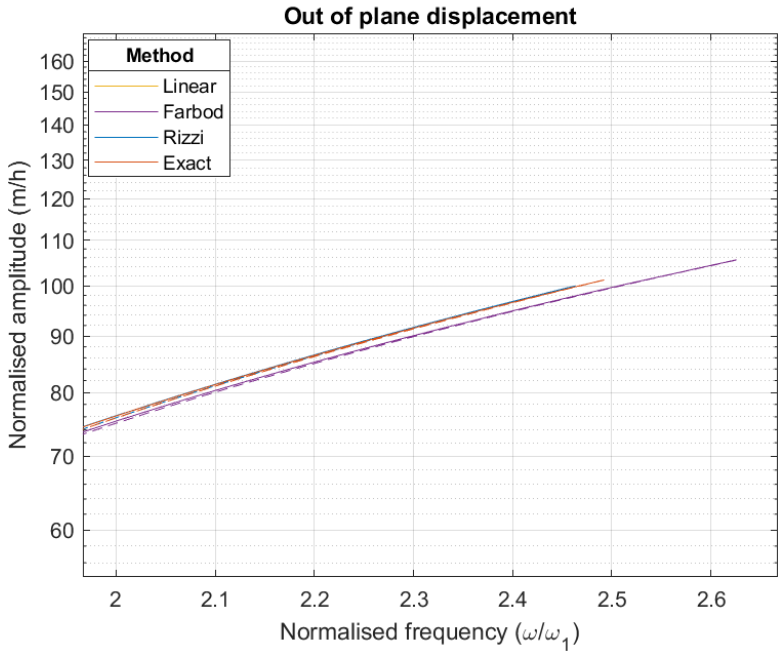


Figure D-8: Frequency response, further into the nonlinear regime: zoom in at tip resonance peaks. $F = 5\text{N}$

In-plane resonance

If the excitation force is further increased to 10N, the in-plane response shows a small resonance at $2.5\omega_1$, which is also $\frac{1}{8}\omega_2$. Note that the exact solution shows a small dip, while Rizzi's solution shows a larger peak. The out-of-plane behaviour remains still unaltered by this in-plane phenomena.

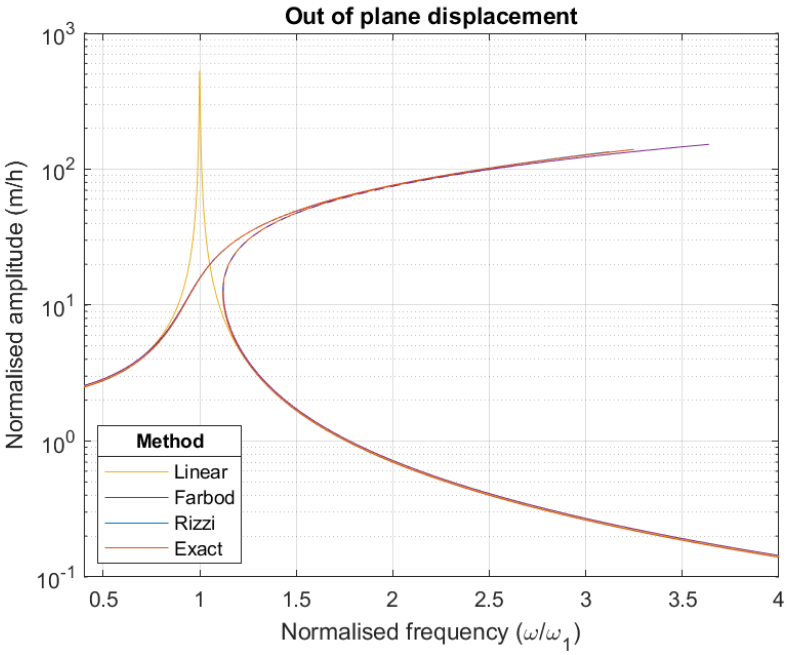


Figure D-9: Frequency response, encountering in-plane resonance at $2.5\omega_1$. $F = 10\text{N}$

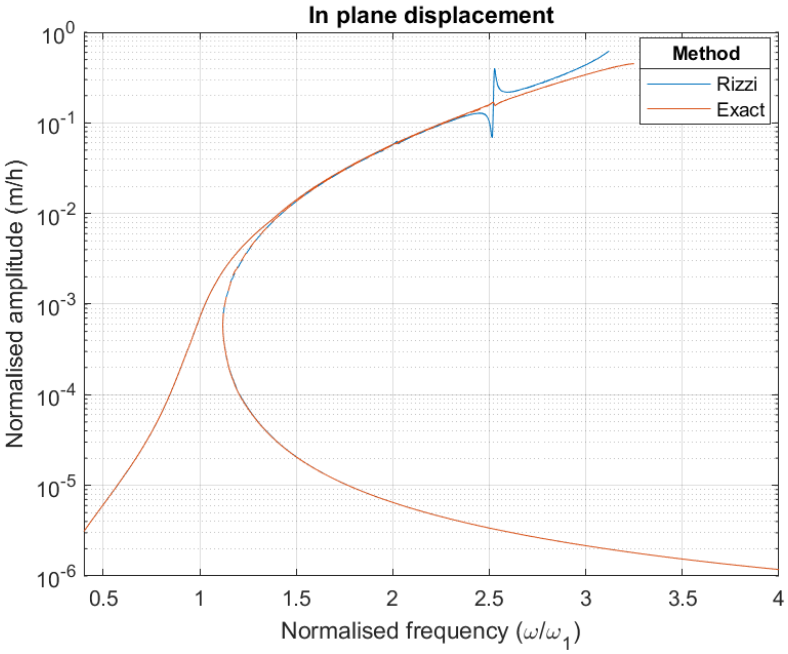


Figure D-10: Frequency response, encountering in-plane resonance at $2.5\omega_1$: In-plane motion. $F = 10\text{N}$

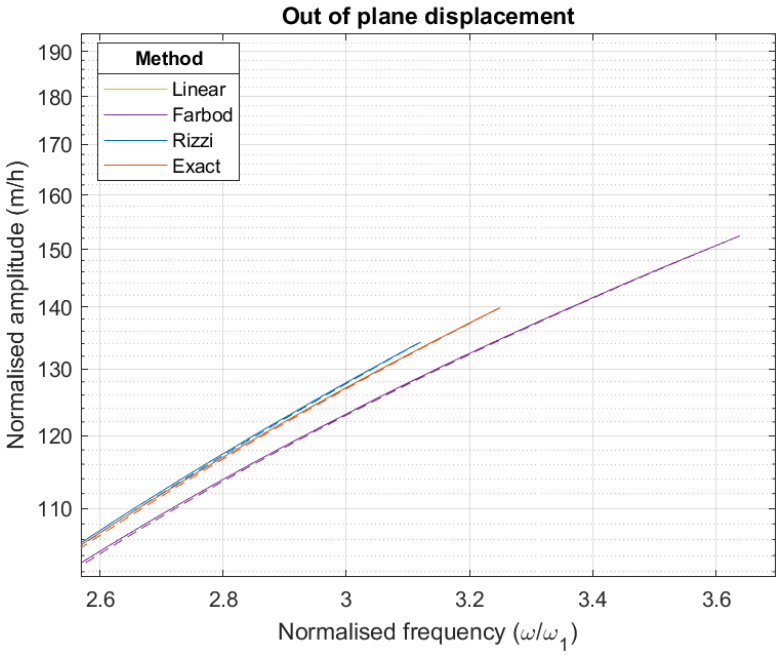


Figure D-11: Frequency response, encountering in-plane resonance at $2.5\omega_1$: zoom in at tip resonance peaks. $F = 10\text{N}$

Locking

When pushing the system far into the nonlinear regime by using a force of $F = 170\text{N}$, the methods show large differences. In-plane resonances occur also at $\omega = 3.3\omega_1 = \frac{1}{6}\omega_2$ and $\omega = 4.93\omega_1 = \frac{1}{4}\omega_2$. When the excitation force is in or decreased, the response does hardly change, as the resonance peak cannot pass the resonance of the in-plane mode (see figure D-12). Farbod's method covers no in-plane dynamics, so it does not see any barriers, yielding a maximum amplitude at way higher frequency. Rizzi's method seems a bit too sensitive for the in-plane resonances, as it founds already at $\omega = 4.6\omega_1$ a barrier which causes it to bend backward.

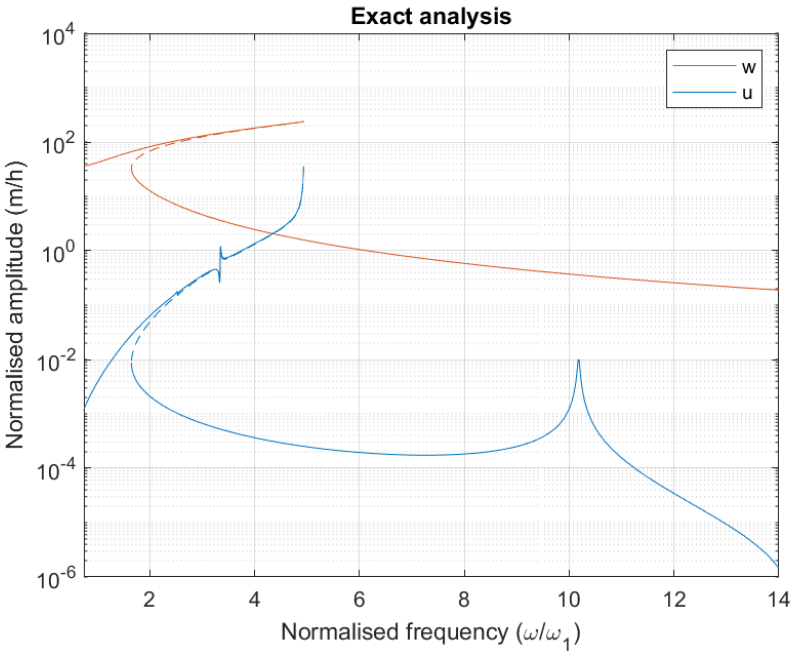


Figure D-12: Frequency response, locked at $4.93\omega_1$. $F = 170\text{N}$

Both Farbod's and Rizzi's method relay on an approximation of the nonlinearity estimated in the equilibrium position. However, at resonance, the out-of-plane amplitude equals about 200 times the string diameter. Therefore, it could be useful to approximate the nonlinearities around this displacement. This method wil be indicated as 'Rizzi's method numerical', or briefly 'RNum'. When the STEP method is applied at the high amplitude region ($w = 230, u = 4$), nonlinear coefficients are somewhat changed. This results in a response which matches the exact solution in this region better then Rizzi's original solution, as it locks at the right frequency. When looking at the in-plane motion, it becomes clear that Rizzi's numerical method is still oversensitive for the in-plane resonances. However, this is hardly influencing the out-of-plane response curve, as it locks at the right frequency.

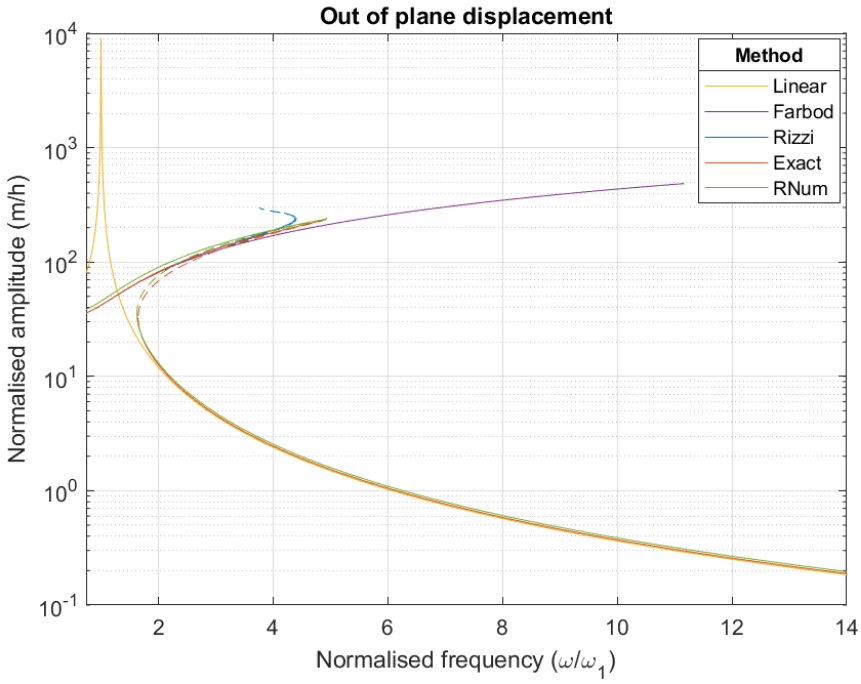


Figure D-13: Frequency response, locked at $4.93\omega_1$. $F = 170\text{N}$

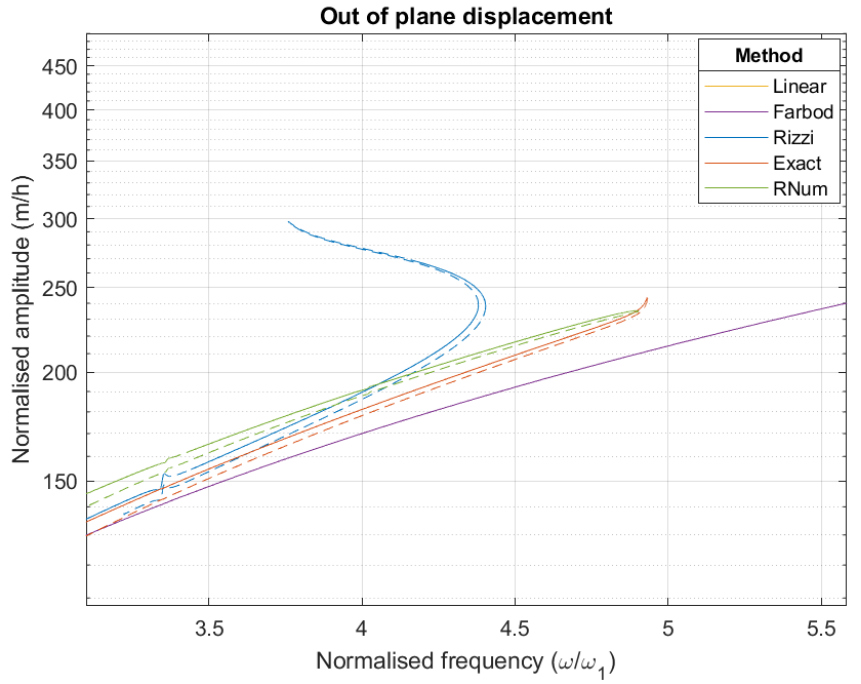


Figure D-14: Frequency response, locked at $4.93\omega_1$: zoom in at tip resonance peaks. $F = 170\text{N}$

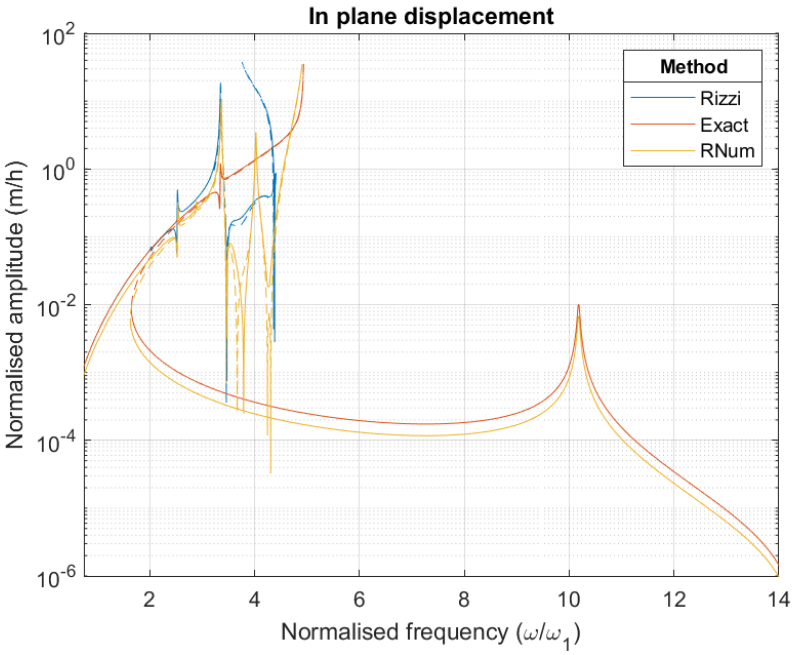


Figure D-15: Frequency response, locked at $4.93\omega_1$: In-plane motion. $F = 170\text{N}$

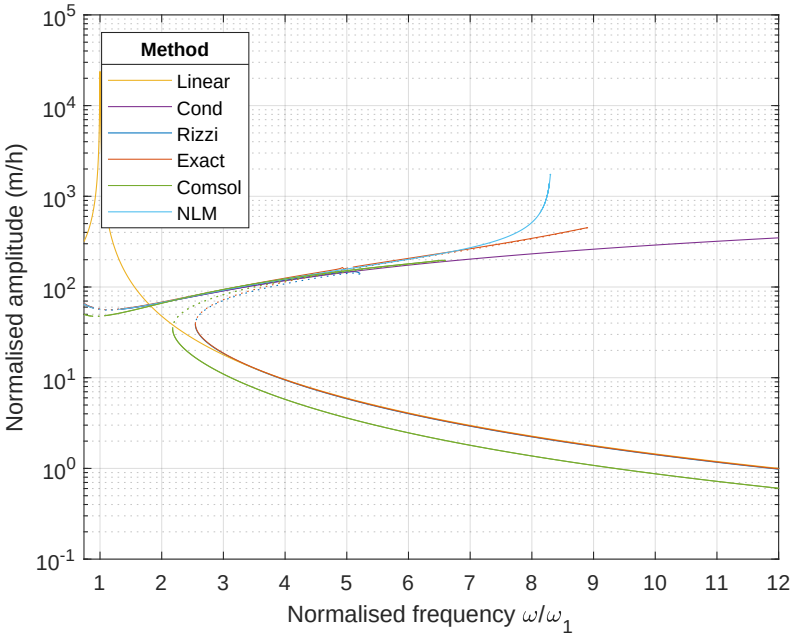


Figure D-16: Frequency response, locked at $4.93\omega_1$: In-plane motion. $F = 170\text{N}$

Parameter choice

The interaction between the in-plane and out-of-plane mode strongly depend on their corresponding eigenfrequencies. By changing the pretension, only the out-of-plane eigenfrequency changes, so the interaction can be altered. When introducing more damping, the sharp peaks in the in-plane response vanish partly, but the overall behaviour does not change.

D-6-3 Conclusions

When considering systems with strongly coupled in- and out-of-plane modes and high vibration amplitude (wrt. structure thickness), both in-plane stiffness and mass can have significant influence on the structure's dynamics. The effect becomes more important as the eigenfrequencies approach each other. In this analysis, Rizzi's method showed the best capability to track the exact solution, as it could be tuned to the range of interest. However, if nonlinear intertia terms would be added to Farbod's method, it could possibly be improved.

D-7 References

[1] A.A. Muravyov, S.A. Rizzi, Computers and Structures 81 (2003) 1513-1523.

D-8 Mathematical derivations

D-8-1 Equations of motion

We obtain for the Lagrangian:

$$\mathcal{L} = E_{kin} - (E_l + E_m) = \frac{M}{2}(\dot{u}^2 + \dot{w}^2) - \frac{1}{2}k(L_l - L_i)^2 - \frac{1}{4}k(L_m - L_i)^2) \tag{D-76}$$

Writing down the Euler-Lagrange equations step by step, using $\frac{dL_l}{du} = \frac{L_0+u}{L_l}$, $\frac{dL_l}{dw} = \frac{w}{L_l}$, $\frac{dL_m}{dw} = 0$ and $\frac{dL_m}{du} = -2$.

$$\frac{\partial \mathcal{L}}{\partial L_m}\frac{\partial L_m}{\partial u} = -\frac{1}{2}k(L_m - L_i)(-2) = k(L_m - L_i) \tag{D-77}$$
$$\frac{\partial \mathcal{L}}{\partial L_l}\frac{\partial L_l}{\partial u} = -k(L_l - L_i)\frac{L_0 + u}{L_l} \tag{D-78}$$
$$\frac{\partial \mathcal{L}}{\partial u} = \frac{\partial \mathcal{L}}{\partial L_l}\frac{\partial L_l}{\partial u} + \frac{\partial \mathcal{L}}{\partial L_m}\frac{\partial L_m}{\partial u} \tag{D-79}$$
$$\frac{\partial \mathcal{L}}{\partial u} = -k(L_l - L_i)\frac{L_0 + u}{L_l} + k(L_m - L_i) \tag{D-80}$$
$$\frac{\partial \mathcal{L}}{\partial w} = \frac{\partial \mathcal{L}}{\partial L_l}\frac{\partial L_l}{\partial w} + \frac{\partial \mathcal{L}}{\partial L_m}\frac{\partial L_m}{\partial w} \tag{D-81}$$
$$\frac{\partial \mathcal{L}}{\partial w} = -k(L_l - L_i)\frac{w}{L_l} \tag{D-82}$$
$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{u}} = M\ddot{u} \tag{D-83}$$
$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{w}} = M\ddot{w} \tag{D-84}$$

Thus we obtain the nonlinear equations of motion:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} &= F_i \\
M\ddot{w} + k(L_l - L_i)\frac{w}{L_l} &= F_y \\
M\ddot{u} + k[(L_l - L_i)\frac{L_0 + u}{L_l} - (L_0 - 2u - L_i)] &= F_x
\end{aligned}$$

Simplified we can write:

$$M\ddot{w} + k(1 - \frac{L_i}{L_l})w = F_y \tag{D-85}$$
$$M\ddot{u} + k[3u + L_i(1 - \frac{L_0 + u}{L_l})] = F_x \tag{D-86}$$

Linearization gives (using $L_l \approx L_0 + u$):

$$M\ddot{u} + 3ku = F_x \tag{D-87}$$
$$M\ddot{w} + kL_t\frac{w}{L_0} = F_y \tag{D-88}$$

D-8-2 Other coefficients

The other coefficients can be determined in a similar way: we start with the displacement field $\mathbf{X}_c = +\phi_2 q_2 = (u = q2, w = 0)$, to obtain the force vector $\mathbf{F}_T = (F_L^y, F_L^x)$. In the linear case we have:

$$\begin{aligned} F_L^y(\phi_2 q_2) &= 0 \\ F_L^x(\phi_2 q_2) &= 3kq_2 \end{aligned}$$

In the nonlinear case we obtain:

$$\begin{aligned} F_L^y(\phi_2 q_2) &= 0 \\ F_L^x(\phi_2 q_2) &= 3kq_2 \end{aligned}$$

Because this is the same as for the nonlinear case, all the coefficients that should appear equal zero:

$$[a_{22}^r] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{D-89}$$

$$[b_{222}^r] = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{D-90}$$

To obtain the last missing coefficients, we apply the displacement field $\mathbf{X}_c = \phi_1 q_1 + \phi_2 q_2 = (w = q1, u = q2)$. In the linear case we have:

$$\begin{aligned} F_L^y(\phi_1 q_1 + \phi_2 q_2) &= k\frac{L_t}{L_0} q_1 \\ F_L^x(\phi_1 q_1 + \phi_2 q_2) &= 3kq_2 \end{aligned}$$

In the nonlinear case, we obtain:

$$\begin{aligned} F_T^y(\phi_1 q_1 + \phi_2 q_2) &= k(1 - \frac{L_i}{\sqrt{(L_0 + q_2)^2 + q_1^2}}) q_1 \\ F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k[3q_2 + L_i(1 - \frac{L_0 + q_2}{\sqrt{(L_0 + q_2)^2 + q_1^2}})] \end{aligned}$$

$$
\begin{aligned}
F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k\left[3q_2 + L_i\left\{1 - \frac{1}{\sqrt{1+\frac{q_1^2}{(L_0+q_2)^2}}}\right\}\right] \\
\frac{1}{\sqrt{1+\frac{q_1^2}{(L_0+q_2)^2}}} &\approx 1 - \frac{1}{2}\frac{q_1^2}{(L_0+q_2)^2} + O(q_2^4) \\
F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k\left[3q_2 + L_i\left\{\frac{1}{2}\frac{q_1^2}{(L_0+q_2)^2}\right\}\right] \\
F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k\left[3q_2 + \frac{1}{2}\frac{L_i}{L_0^2}q_1^2\left\{\frac{1}{(1+\frac{q_2}{L_0})^2}\right\}\right] \\
\frac{1}{(1+\frac{q_2}{L_0})^2} &\approx 1 - 2\frac{q_2}{L_0} + O(q_2^2) \\
F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k\left[3q_2 + \frac{1}{2}\frac{L_i}{L_0^2}q_1^2\left\{1 - 2\frac{q_2}{L_0}\right\}\right] \\
F_T^x(\phi_1 q_1 + \phi_2 q_2) &= k\left[3q_2 + \frac{1}{2}\frac{L_i}{L_0^2}q_1^2 - \frac{L_i}{L_0^3}q_1^2 q_2\right]
\end{aligned}
$$

For the vertical force, we have:

$$
\begin{aligned}
F_T^y(\phi_1 q_1 + \phi_2 q_2) &= kq_1(1 - \frac{L_i}{\sqrt{(L_0+q_2)^2+q_1^2}}) \\
F_T^y(\phi_1 q_1 + \phi_2 q_2) &= kq_1(1 - \frac{L_i}{L_0+q_2}\frac{1}{\sqrt{1+\frac{q_1^2}{(L_0+q_2)^2}}}) \\
F_T^y(\phi_1 q_1 + \phi_2 q_2) &\approx kq_1(1 - \frac{L_i}{L_0+q_2}\{1 - \frac{1}{2}\frac{q_1^2}{(L_0+q_2)^2}\}) \\
F_T^y(\phi_1 q_1 + \phi_2 q_2) &\approx k(q_1 - q_1\frac{L_i}{L_0+q_2} + \frac{1}{2}L_i\frac{q_1^3}{(L_0+q_2)^3}) \\
F_T^y(\phi_1 q_1 + \phi_2 q_2) &\approx k(q_1\frac{L_0+q_2-L_i}{L_0+q_2} + \frac{1}{2}\frac{L_i}{L_0^3}q_1^3\frac{1}{(L_0+q_2)^3}) \\
F_T^y(\phi_1 q_1 + \phi_2 q_2) &\approx k(q_1\frac{L_t+q_2}{L_0}\frac{1}{1+\frac{q_2}{L_0}} + \frac{1}{2}\frac{L_i}{L_0^3}q_1^3 + O(q_1^3 q_2^2))
\end{aligned}
$$

$$\begin{aligned}
\frac{1}{(1+\frac{q_2}{L_0})} \approx & \quad 1-\frac{q_2}{L_0}+\frac{q_2^2}{L_0^2}+O(q_2^3)\\
F_T^y(\phi_1 q_1+\phi_2 q_2) \approx & \quad k(q_1\frac{L_t+q_2}{L_0}\{1-\frac{q_2}{L_0}+\frac{q_2^2}{L_0^2})\}+\frac{1}{2}\frac{L_i}{L_0^3}q_1^3)\\
F_T^y(\phi_1 q_1+\phi_2 q_2) \approx & \quad k(q_1\{\frac{L_t}{L_0}+\frac{q_2}{L_0}-q_2\frac{L_t}{L_0^2}-\frac{q_2^2}{L_0^2}+q_2^2\frac{L_t}{L_0^3})\}+\frac{1}{2}\frac{L_i}{L_0^3}q_1^3+O(q_1^3q_2))\\
\frac{q_2}{L_0}(1-\frac{L_t}{L_0}) = & \quad \frac{q_2}{L_0}(\frac{-L_0+(L_0-L_i)}{L_0})=q_2\frac{L_i}{L_0^2}\\
\frac{q_2^2}{L_0^2}(-1+\frac{L_t}{L_0}) = & \quad -q_2^2\frac{L_i}{L_0^3}\\
F_T^y(\phi_1 q_1+\phi_2 q_2) \approx & \quad k\frac{L_t}{L_0}q_1+k\frac{L_i}{L_0^2}q_1q_2-k\frac{L_i}{L_0^3}q_1q_2^2+\frac{1}{2}\frac{L_i}{L_0^3}q_1^3
\end{aligned}$$

The coefficients can be found using the procedure earlier decribed. We find:

$$[a_{12}^r]=\begin{pmatrix} k\frac{L_i}{L_0^2}\\ 0\end{pmatrix} \tag{D-91}$$

$$[b_{112}^r]=\begin{pmatrix} 0\\ -k\frac{L_i}{L_0^3}\end{pmatrix} \tag{D-92}$$

$$[b_{122}^r]=\begin{pmatrix} -k\frac{L_i}{L_0^3}\\ 0\end{pmatrix} \tag{D-93}$$

D-8-3 Method Muravyov and Rizzi

According to the paper we impose a static displacement $\mathbf{X}_c=+\phi_1 q_1=(u=0,w=q_1)$ on the structure. The static force can be obtained from the equations of motion, by setting the accelerations equal to zero. For the linear case, this results in a force vector:

$$\mathbf{F}_L(\phi_1 q_1)=(0,k\frac{L_t}{L_0}q_1) \tag{D-94}$$

For the nonlinear case this results in:

$$\mathbf{F}_T(+\phi_1 q_1)=k[3u+L_i(1-\frac{L_0+u}{L_l})],k(1-\frac{L_i}{L_l})w) \tag{D-95}$$

$$\mathbf{F}_T(+\phi_1 q_1)=(k(L_i-\frac{L_i}{\sqrt{1+\frac{q_1^2}{L_0^2}}}),k(L_0-\frac{L_i}{\sqrt{1+\frac{q_1^2}{L_0^2}}})\frac{q_1}{L_0}) \tag{D-96}$$

The term with the square root can be estimated using a Taylor approximation:

$$\frac{1}{\sqrt{1+x^2}} \approx 1 - \frac{1}{2}x^2 + \frac{3}{8}x^4 + O(x^6) \tag{D-97}$$

$$\mathbf{F}_T(+\phi_1 q_1) \approx (\frac{1}{2}k\frac{L_i}{L_0^2}q_1^2 - \frac{3}{8}k\frac{L_i}{L_0^4}q_1^4 + O(q_1^6), k\frac{L_t}{L_0}q_1 + \frac{1}{2}k\frac{L_i}{L_0^3}q_1^3 - \frac{3}{8}k\frac{L_i}{L_0^5}q_1^5 + O(q_1^7)) \tag{D-98}$$

D-8-4 Equations of motion

Now the equations of motion of the approximate model (containing max cubic order terms) can be written:

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + a_{12}^{(1)}q_1q_2 + b_{111}^{(1)}q_1^3 + b_{122}^{(1)}q_1q_2^2 = F^y \tag{D-99}$$

$$m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 + b_{112}^{(2)}q_1^2q_2 = F^x \tag{D-100}$$

From this we can see that the in-plane eigenmode, has no source of nonlinearity from itself ($a_{22}^{(2)}$ and $b_{222}^{(2)}$ equal zero), it has only nonlinear coupling to the out-of-plane mode. The quadratic term in the equation for out-of-plane motion ($(a_{12}^{(1)})$ indicates softening behaviour, lowering the effective stiffness of this mode. If we consider first the static case with no external forcing on the second mode, we can write q_2 in terms of q_1 and substitute this back in the equation of motion for q_1, yielding the effective stiffness of this mode.

$$k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 + b_{112}^{(2)}q_1^2q_2 = 0$$

$$q_2 = \frac{-a_{11}^{(2)}q_1q_1}{k_1^{(2)} + b_{112}^{(2)}q_1^2}$$

Substitute $q_2(q_1)$ in the static equation for q_1:

$$k_1^{(1)}q_1 + a_{12}^{(1)}q_1\frac{-a_{11}^{(2)}q_1q_1}{k_1^{(2)} + b_{112}^{(2)}q_1^2} + b_{111}^{(1)}q_1^3 + b_{122}^{(1)}q_1\left\{\frac{-a_{11}^{(2)}q_1q_1}{k_1^{(2)} + b_{112}^{(2)}q_1^2}\right\}^2 = F^y$$

$$k_1^{(1)}q_1 - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}q_1^3\frac{1}{1+\frac{b_{112}^{(2)}}{k_1^{(2)}}q_1^2} + b_{111}^{(1)}q_1^3 + b_{122}^{(1)}O(q_1^5) = F^y$$

$$k_1^{(1)}q_1 - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}q_1^3(1+O(q_1^2)) + b_{111}^{(1)}q_1^3 + b_{122}^{(1)}O(q_1^5) = F^y$$

$$k_1^{(1)}q_1 - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}q_1^3 + b_{111}^{(1)}q_1^3 = F^y$$

Use $k_1^{(2)} = 3k$, $a_{11}^{(2)} = \frac{1}{2}k\frac{L_i}{L_0^2}$, $a_{12}^{(1)} = 2a_{11}^{(2)} = k\frac{L_i}{L_0^2}$

$$b_{111-eff}^{(1)} = b_{111}^{(1)} - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}} = \frac{1}{2}k\frac{L_i}{L_0^3} - \frac{1}{3k}\frac{1}{2}\left\{k\frac{L_i}{L_0^2}\right\}^2 = \frac{1}{2}k\frac{L_i}{L_0^3} - \frac{1}{6}k\frac{L_i^2}{L_0^4} \tag{D-101}$$

Use $L_i \approx L_0$

$$b_{111-eff}^{(1)} = \frac{1}{3}k\frac{L_i}{L_0^3} \tag{D-102}$$

D-8-5 Nonlinear mass

The in-plane mode has a nonzero modal mass, so at some point this may influence the dynamic response of the out-of-plane mode. If we consider harmonic motion around the first eigenfrequency (of the out-of-plane mode), the in-plane mode will still follow it's forcing terms, as it's mass has small influence at $\omega = \omega_1 << \omega_2$. To investigate the lowest-order influence, we neglect the b_{112} and b_{122} stiffness, as they result in higher order coupling. So we obtain the following simplified equations of motion:

$$\begin{aligned} m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + a_{12}^{(1)}q_1q_2 + b_{111}^{(1)}q_1^3 &= \sin(\omega_f t) && \text{(D-103)} \\ m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 + a_{11}^{(2)}q_1q_1 &= 0 && \text{(D-104)} \end{aligned}$$

Consider motion of the out-of-plane mode following the forcing term:

$$\begin{aligned} q_1 &= \hat{q_1}\sin(\omega_f t) \\ q_1^2 &= \hat{q_1}^2\sin(\omega_f t)^2 = \hat{q_1}^2\frac{1}{2}(1-\cos(2\omega_f t)) \end{aligned}$$

We can consider the quadratic coupling term in the in-plane equation of motion as an external forcing, yielding:

$$m^{(2)}\ddot{q_2} + k_1^{(2)}q_2 = -a_{11}^{(2)}q_1q_1 = -a_{11}^{(2)}\hat{q_1}^2\frac{1}{2}(1-\cos(2\omega_f t))$$

The static force yields a static displacement:

$$q_{2-\text{static}} = -\frac{a_{11}^{(2)}}{2k_1^{(2)}}\hat{q_1}^2 = -\beta\frac{1}{2}\hat{q_1}^2 \tag{D-105}$$

The dynamic force can be analysed using Laplace (only looking at steady state solutions) (use $\omega_{2f} = 2\omega_f$):

$$m^{(2)}\ddot{q}_2 + k_1^{(2)} q_2 = a_{11}^{(2)} \hat{q}_1^2 \frac{1}{2}\cos(\omega_{2f} t) = f(\omega_{2f}) \tag{D-106}$$

$$Q_2(-m^{(2)}(\omega_{2f})^2 + k_1^{(2)}) = F(\omega_{2f}) \tag{D-107}$$

$$\frac{Q_2(\omega_{2f})}{F(\omega_{2f})} = \frac{1}{(k_1^{(2)} - m^{(2)}(\omega_{2f})^2} = \frac{1}{k_1^{(2)}} \frac{1}{1 - (\frac{\omega_{2f}}{\omega_2})^2} \tag{D-108}$$

$$q_{2-\text{dynamic}} = \frac{a_{11}^{(2)}}{2k_1^{(2)}} \hat{q}_1^2 \frac{1}{1 - (\frac{2\omega_f}{\omega_2})^2} \cos(2\omega_f t) = \frac{1}{2}\beta\alpha(\omega_f)\hat{q}_1^2 \cos(2\omega_f t) \tag{D-109}$$

The total solution then writes:

$$q_2 = q_{2-\text{static}} + q_{2-\text{dynamic}} = -\beta \frac{1}{2}\hat{q}_1^2 + \frac{1}{2}\beta\alpha(\omega_f)\hat{q}_1^2 \cos(2\omega_f t) \tag{D-110}$$

$$q_2 = -\beta\hat{q}_1^2 \frac{1}{2}(1 - \alpha\cos(2\omega_f t)) = -\beta\hat{q}_1^2 \left[\frac{1}{2}\alpha(1 - \cos(2\omega_f t)) - \frac{1}{2}(\alpha - 1)\right] \tag{D-111}$$

$$q_2 = -\beta\alpha\hat{q}_1^2 \frac{1}{2}\left[(1 - \cos(2\omega_f t)) - (1 - \frac{1}{\alpha})\right] \tag{D-112}$$

Recall

$$q_1^2 = \hat{q}_1^2 \sin(\omega_f t)^2 = \hat{q}_1^2 \frac{1}{2}(1 - \cos(2\omega_f t))$$

$$q_2 = -\beta\alpha\left[q_1^2 - \hat{q}_1^2 \frac{1}{2}(1 - \frac{1}{\alpha})\right] = -\beta\alpha\left[q_1^2 - \epsilon\right] \tag{D-113}$$

For $\omega_f = 0$, $\epsilon = 0$, so $q_2 \geq 0$. For $\omega_f < 2\omega_n$, $\alpha \geq 1$, so in this domain $0 < \epsilon < \frac{1}{2}\hat{q}_1^2$. Actually, ϵ is translating the squared sine downward. This causes q_2 to also take negative values, meaning that the in-plane mode is also moving towards the centre.

This can be inserted in the out-of-plane equation of motion:

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \left(-\beta\alpha\left[q_1^2 - \epsilon\right]\right) + b_{111}^{(1)} q_1^3 = \sin(\omega_f t) \tag{D-114}$$

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \epsilon - \beta\alpha a_{12}^{(1)} q_1^3 + b_{111}^{(1)} q_1^3 = \sin(\omega_f t) \tag{D-115}$$

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + a_{12}^{(1)} q_1 \epsilon + q_1^3 (b_{111}^{(1)} - \beta\alpha a_{12}^{(1)}) = \sin(\omega_f t) \tag{D-116}$$

As ϵ depends on $\hat{q}_1$ which is not analythic available, ϵ is removed. However, it is assumed to lower the maximum value of q_2. As q_2 seems to have a softening effect on q_1, this softening can be seen as an upper bound, so probably the amplitude of q_1 will be slightly lower. When putting back all the original terms in the equation,

$$m^{(1)}\ddot{q}_1 + k_1^{(1)} q_1 + q_1^3 \left(b_{111}^{(1)} - \frac{a_{11}^{(2)} a_{12}^{(1)}}{k_1^{(2)}} \frac{1}{1 - (\frac{2\omega_f}{\omega_2})^2}\right) = \sin(\omega_f t) \tag{D-117}$$

Here, we find again the same factor we found for the static analysis $\frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}$. However, an additional term governing the in-plane inertia is now visible. This term becomes infinite at half the in-plane eigenfrequency, removing the cubic hardening effect of the out-of-plane mode itself. Therefore, the backbone curve will not pass this point.

Small in-plane damping (eg $Q > 100$) can be added, as long as the phase shift is small enough to neglect.

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + q_1^3\left(b_{111}^{(1)} - \frac{a_{11}^{(2)}a_{12}^{(1)}}{k_1^{(2)}}\frac{1}{\sqrt{(1-(\frac{2\omega_f}{\omega_2})^2)^2 + (2\zeta\frac{2\omega_f}{\omega_2})^2}}\right) = \sin(\omega_f t) \tag{D-118}$$

Damping can reduce the softening influence of the inertia, and can prevent the out-of-plane to lock when the in-plane mode starts to resonate. The above derivation is not exact, but it contains the features regarding strong mode interaction and reveals the dependencies.

One interesting extension could be to include in the in-plane equation the cubic coupling term $b_{112}^{(2)}q_1^2q_2$. This yields a linear stiffness which is parametrically excited, resulting in resonances at integer fractions like $2\omega_f = \frac{1}{n}\omega_2, n = 2, 3, 4...$ (as described in [17]).

From this we can see that around the out-of-plane eigenfrequency, the in-plane inertia does not play a big role. However, at higher frequency, the dynamic stiffness of the in-plane motion reduces, by which the negative cubic correction term increases. At $\omega_f = \frac{1}{2}\omega_2$, the in-plane mode should resonate and the out-of-plane mode could get negative cubic stiffness creating instability.

To compare this with other ways of modeling nonlinear inertia, we approximate the fraction with a Taylor series, which is valid for small $\frac{\omega_f}{\omega_2}$:

$$\frac{1}{1-(\frac{2\omega_f}{\omega_2})^2} \approx 1 + \left(\frac{2\omega_f}{\omega_2}\right)^2 + O(\left(\frac{2\omega_f}{\omega_2}\right)^4) \tag{D-119}$$

Yielding

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 - a_{12}^{(1)}q_1\frac{a_{11}^{(2)}}{k_1^{(2)}}\hat{q}_1^2\left(1 + \left(\frac{2\omega_f}{\omega_2}\right)^2\right) + b_{111}^{(1)}q_1^3 = \sin(\omega_f t) \tag{D-120}$$

Split up the quadratic correction term in a static and dynamic part:

$$m^{(1)}\ddot{q_1} + k_1^{(1)}q_1 + b_{111-eff}^{(1)}q_1^3\left[1 - \frac{1}{2}\left(\frac{2\omega_f}{\omega_2}\right)^2\right] = \sin(\omega_f t) \tag{D-121}$$

The nonlinear intertia appears to scale with $q_1^3\omega^2$. Terms like $x\dot{x}^2$ or $\ddot{x}x^2$ (which could be introduced to the out-of-plane equation of motion), have the same scaling.

Appendix E

Supplementary material for Chapter 3: Tables of coefficients

Table E-1: Non-dimensionalised linear modal values of a **string**. (Extended version of Table 3-2.) Dimensional pre-factors are listed in Table 3-3

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
m/m_t	0.5	0.5	0.5	0.5	0.499	0.499	0.499
k/k_0	4.93	19.7	44.4	78.9	123	178	242
ω/ω_0	1	2	3	4	5	6	7

Table E-2: Non-dimensionalised linear modal values of a **string incl in-plane modes**. (Extended version of Table 3-7.) Dimensional pre-factors are listed in Table 3-6, although eigenfrequencies are normalised w.r.t. the fundamental mode (ω_1). Note that Eq. 4, Eq. 5 & Eq. 6 describe in-plane modes.

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6
m/m_t	0.5	0.5	0.499	0.5	0.501	0.497
k/k_0	4.93	19.7	44.4	4.97	19.9	44.6
ω/ω_1	1	2	3	23.6	47.2	70.9

Table E-3: Non-dimensionalised linear modal values of a **circular membrane**. (Extended version of Table 3-11.) Dimensional pre-factors are listed in Table 3-12 Modes: 1,2,4,6,7,9,11.

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
m/m_t	0.269	0.24	0.25	0.115	0.233	0.133	0.227
k/k_0	4.9	11.1	20.8	11.1	29.8	20.6	41.2
ω/ω_0	1	1.59	2.14	2.3	2.66	2.92	3.16

Table E-4: Non-dimensionalised linear modal values of a **square membrane**. (Extended version of Table 3-17.) Dimensional pre-factors are listed in Table 3-18 Modes: 1,2,4,5,7,9,11.

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
m/m_t	0.25	0.215	0.25	0.206	0.154	0.227	0.248
k/k_0	4.93	10.6	19.7	20.3	19.8	38.2	44.2
ω/ω_0	1	1.58	2	2.24	2.55	2.92	3

Table E-5: Non-dimensionalised linear modal values of **A. Keşkekler's membrane** shown in section 3-3-2. Dimensional pre-factors are listed in Table 3-12, although eigenfrequencies are normalised w.r.t. the fundamental mode (ω_1). Modes: 1,2,4,6,7,9,15

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
m/m_t	0.204	0.195	0.183	0.103	0.192	0.118	0.0698
k/k_0	5.78	16.5	34.6	24.2	68.3	55.4	64.9
ω/ω_1	1	1.73	2.59	2.89	3.54	4.07	5.73

Table E-6: Non-dimensionalised linear modal values of an **elliptic membrane**, $R_2 = 2R$. Dimensional pre-factors are listed in Table 3-12, although eigenfrequencies are normalised w.r.t. the fundamental mode (ω_1). Modes: 1-7

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
m/m_t	0.25	0.257	0.226	0.201	0.193	0.232	0.171
k/k_0	5.61	10.1	14.2	14.9	18.1	23.2	22.6
ω/ω_1	1	1.33	1.68	1.82	2.05	2.12	2.42

Table E-7: Ext. version of Table 3-5, coeff. for a **string**. Error: $\epsilon_{\max} < 20\%$, $\epsilon_{\text{mean}} < 4\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
b_{111}	12.2	0	0.171	0	0.003 04	0	−0.0022
b_{112}	0	49.8	0	1.46	0	0.051	0
b_{113}	0.608	0	112	0	2.65	0	0.0137
b_{114}	0	1.4	0	199	0	4.19	0
b_{115}	0.003 78	0	2.6	0	310	0	6.07
b_{116}	0	0.0056	0	4.15	0	447	0
b_{117}	−0.001 06	0	0.009 89	0	6.04	0	608
b_{122}	49.9	0	2.38	0	3.66	0	0.0293
b_{123}	−0.343	4.43	−1.15	8.37	−0.272	13.5	−0.626
b_{124}	3.45	−0.365	8.84	−1.66	14.7	−0.609	20.5
b_{125}	−0.798	8.48	−0.862	15.2	−4.8	22.4	−1.71
b_{126}	0.287	−1.21	13.4	−1.51	22.5	−6	31.2
b_{127}	0.143	−0.596	−0.805	20.3	−1.45	31.1	−8.14
b_{133}	112	0	1.12	0	7.99	0	11.2
b_{134}	−0.0343	7.98	1.51	−1.27	1.12	24.9	−0.564
b_{135}	5.91	0	16.1	0	−0.653	0	36.7
b_{136}	0.031	12.8	−1.49	24.7	−0.433	−2.76	0.271
b_{137}	−0.794	0	21.4	0	35.4	0	−2.89
b_{144}	200	0	0.711	0	0.0884	0	19.6
b_{145}	0.188	14.6	1.35	2.88	4.23	2.39	1.88
b_{146}	10.2	0.181	25.4	0.355	0.233	2.97	0.032
b_{147}	0.711	19.4	0.063	37.4	1.55	−0.727	7.24
b_{155}	312	0	1.11	0	0.0555	0	0.152
b_{156}	−0.171	21.2	−0.682	2.37	−4.22	6.12	−3.12
b_{157}	14.9	0	37.1	0	0.536	0	3.83
b_{166}	449	0	1.6	0	0.027	0	−0.0186
b_{167}	0.17	29.5	−0.053	−0.752	−3.12	−6.1	−8.26
b_{177}	611	0	2.17	0	0.0351	0	−0.0775
b_{222}	0	196	0	0	0	2.28	0
b_{223}	1.75	0	446	0	0.013	0	12.3
b_{224}	0	0	0	792	0	0	0
b_{225}	2.86	0	0.005 89	0	1.24×10^3	0	0.000 294
b_{226}	0	7.16	0	0	0	1.78×10^3	0
b_{227}	0.002 72	0.000 128	11.9	−0.000 14	0.001 04	0	2.42×10^3
b_{233}	0	446	0	10.5	0	0.398	0
b_{234}	7.01	0.334	20.6	1.65	35.1	−0.411	−0.855
b_{235}	0.307	−0.631	−1.47	35.2	4.18	53.2	−1.76
b_{236}	10.7	1.19	2.28	0.487	53.1	5.94	75.1
b_{237}	0.439	27	4.1	0.451	0.0871	74.1	10.7
b_{244}	0	792	0	0	0	28.4	0
b_{245}	11.4	0.64	34.4	−2.62	−4.02	−0.522	81.6
b_{246}	0	−2.69	0	60.8	0	−7.52	0
b_{247}	15.9	−0.66	−0.884	2.62	81.2	−0.471	−8.03
b_{255}	0	1.24×10^3	0	0.0394	0	1.1	0
b_{256}	17.5	−1.19	52	−1.07	6.32	−5.93	1.72
b_{257}	0.202	−0.667	−1.84	80.7	4.17	0.163	−8.18
b_{266}	0	1.78×10^3	0	0	0	4.5	0
b_{267}	24.1	1.19	73.2	0.0757	1.69	5.93	12.4
b_{277}	0	2.42×10^3	0	0.000 354	0	2.16	0.000 741
b_{333}	0.004 55	0	990	0	0.107	0	0.314
b_{334}	0	9.96	0	1.78×10^3	0	0.0111	0
b_{335}	6.18	0	0.007 19	0	2.78×10^3	0	0.0132
b_{336}	0	0	0	0.000 223	0	4×10^3	0
b_{337}	8.6	0	0.0227	−0.000 308	0.0147	0.000 13	5.44×10^3
b_{344}	0.004 34	0	1.78×10^3	0	33.2	0	0.0186
b_{345}	−0.211	33.3	−0.378	65.8	−2.26	99.7	0.834
b_{346}	20.1	−0.156	−1.46	2.58	99.7	−5.92	141
b_{347}	−0.61	0.394	−1.5	2.71	−0.735	140	−8.17
b_{355}	0.006 68	0	2.78×10^3	0	0.0295	0	72.1
b_{356}	0.222	49.9	−1.48	98.7	4.08	−5.86	0.609
b_{357}	28.9	0	−3.99	0	151	0	−10.5
b_{366}	0.000 541	0	4×10^3	0	0.0147	0	0.0421
b_{367}	0.327	69.4	1.48	139	1.56	6	8.05
b_{377}	0.000 825	0	5.45×10^3	0	0.0809	−0.000 283	0.175
b_{444}	0	0	0	3.13×10^3	0	0	0.000 116
b_{445}	0	0	32.5	0	4.94×10^3	0	0.0171
b_{446}	0	25.9	0	0	0	7.11×10^3	0
b_{447}	15.1	0.000 174	0.001 01	−0.001 64	0.002 78	0.000 291	9.68×10^3
b_{455}	0	0.009 81	0	4.94×10^3	0	81.2	0
b_{456}	0.000 673	0.504	97.6	0.341	162	2.95	228
b_{457}	0.293	76.3	0.261	−2.59	−4.03	228	8.05
b_{466}	0	0	0	7.11×10^3	0	0	0
b_{467}	−0.288	−0.503	137	2.63	226	−5.86	−8
b_{477}	0	0.0192	0	9.68×10^3	−0.000 303	−0.000 856	−0.003 26
b_{555}	0	0	0	0	7.63×10^3	0	−0.000 158
b_{556}	0	−0.000 26	0	80.3	0	1.11×10^4	0
b_{557}	0	0.000 246	70.1	−0.000 824	0.0133	0.000 246	1.51×10^4
b_{566}	0	0	0.0163	0	1.11×10^4	0	169
b_{567}	0	−0.000 865	−0.878	225	−0.224	336	−3.72
b_{577}	0.007 26	0	0.000 528	−0.000 301	1.51×10^4	0	0.0181
b_{666}	0	−0.044	0	0	0.000 126	1.58×10^4	−0.000 211
b_{667}	−0.000 238	0.000 407	0.002 46	−0.0014	168	0.001 81	2.18×10^4
b_{677}	0	−0.000 578	−0.000 273	0.0246	0	2.18×10^4	0.002 69
b_{777}	−0.029	0.0851	0.497	−0.266	1.4	0.128	2.93×10^4

Table E-8: Ext. version of Table 3-9, coeff. for a **string incl in-plane modes**. Note that Eq. 4, Eq. 5 & Eq. 6 describe in-plane modes. Error: $\epsilon_{\max} < 0.2\%$, $\epsilon_{\text{mean}} < 0.004\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6
b_{111}	18.3	0	18.3	0	0	0
b_{112}	0	146	0	0	0	0
b_{113}	54.8	0	328	0	0	0
b_{114}	0	0	0	48.7	−48.8	0.005 33
b_{115}	0	0	0	−48.8	195	146
b_{116}	0	0	0	0.004 03	146	437
b_{122}	146	0	219	0	0	0
b_{123}	0	438	0	0	0	0
b_{124}	0	0	0	0	0	0
b_{125}	0	0	0	0	0	0
b_{126}	0	0	0	0	0	0
b_{133}	328	0	−0.0227	0	0	0
b_{134}	0	0	0	146	−292	−437
b_{135}	0	0	0	−292	−0.006 55	876
b_{136}	0	0	0	−437	876	−0.0632
b_{144}	48.7	0	73	0	0	0
b_{145}	−97.5	0	−292	0	0	0
b_{146}	0.007 47	0	−437	0	0	0
b_{155}	195	0	−0.003 86	0	0	0
b_{156}	292	0	876	0	0	0
b_{166}	437	0	−0.0141	0	0	0
b_{222}	0	292	0	0	0	0
b_{223}	219	0	1.31×10^3	0	0	0
b_{224}	0	0	0	292	0	−292
b_{225}	0	0	0	0	781	0
b_{226}	0	0	0	−292	0	1.75×10^3
b_{233}	0	1.31×10^3	0	0	0	0
b_{234}	0	0	0	0	0	0
b_{235}	0	0	0	0	0	0
b_{236}	0	0	0	0	0	0
b_{244}	0	292	0	0	0	0
b_{245}	0	0	0	0	0	0
b_{246}	0	−583	0	0	0	0
b_{255}	0	781	0	0	0	0
b_{256}	0	0	0	0	0	0
b_{266}	0	1.75×10^3	0	0	0	0
b_{333}	0.004 17	0	1.48×10^3	0	0	0
b_{334}	0	0	0	438	−439	0.0182
b_{335}	0	0	0	−439	1.76×10^3	−0.0141
b_{336}	0	0	0	0.006 27	0.001 05	3.93×10^3
b_{344}	73	0	438	0	0	0
b_{345}	−292	0	−877	0	0	0
b_{346}	−437	0	0.0373	0	0	0
b_{355}	0.000 582	0	1.76×10^3	0	0	0
b_{356}	876	0	−0.0161	0	0	0
b_{366}	−0.011	0	3.93×10^3	0	0	0
b_{444}	0	0	0	292	0	−290
b_{445}	0	0	0	0	2.34×10^3	0
b_{446}	0	0	0	−875	0	5.24×10^3
b_{455}	0	0	0	2.34×10^3	0	-3.51×10^3
b_{456}	0	0	0	0	-7.02×10^3	0
b_{466}	0	0	0	5.24×10^3	0	0.502
b_{555}	0	0	0	0	4.7×10^3	0
b_{556}	0	0	0	-3.51×10^3	0	2.1×10^4
b_{566}	0	0	0	0	2.1×10^4	0
b_{666}	0	0	0	0.132	0	2.35×10^4

Table E-9: Ext. version of Table 3-8, quadratic coeff. for a **string incl in-plane modes**. Note that Eq. 4, Eq. 5 & Eq. 6 describe in-plane modes.

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6
a_{11}	0	0	0	7.75	−0.000 153	−0.0014
a_{12}	0	0	0	0	0	0
a_{13}	0	0	0	46.5	−93.1	0.006 46
a_{14}	15.5	0	46.5	0	0	0
a_{15}	−0.000 152	0	−93.1	0	0	0
a_{16}	−0.001 32	0	0.008 06	0	0	0
a_{22}	0	0	0	0	−62.1	0
a_{23}	0	0	0	0	0	0
a_{24}	0	0	0	0	0	0
a_{25}	0	−124	0	0	0	0
a_{26}	0	0	0	0	0	0
a_{33}	0	0	0	0.000 349	−0.001 73	−209
a_{34}	46.5	0	−0.001 16	0	0	0
a_{35}	−93.1	0	−0.001 73	0	0	0
a_{36}	−0.001 08	0	−418	0	0	0
a_{44}	0	0	0	0	−186	0
a_{45}	0	0	0	−372	0	1.12×10^3
a_{46}	0	0	0	0	1.12×10^3	0
a_{55}	0	0	0	0	0	0
a_{56}	0	0	0	1.12×10^3	0	0.0162
a_{66}	0	0	0	0	−0.0111	0

Table E-10: Coeff. for a **circ. membrane** (as Table 3-13). Modes: 1,2,4,6,7,9,11. Error: $\epsilon_{\max} < 1.3\%$, $\epsilon_{\text{mean}} < 0.26\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
b_{111}	2.84	0	−0.000 122	−0.57	0	0	0
b_{112}	0	9.26	−0.000 107	0.000 187	−0.000 951	1.79	0.000 884
b_{113}	−0.000 357	−0.000 105	12.4	0.000 777	0.000 174	0.001 59	0.000 771
b_{114}	−1.71	0.000 174	0.000 754	9.27	−0.000 657	0.000 617	0.000 131
b_{115}	−0.000 275	−0.000 181	0.000 18	−0.000 635	14.7	−0.001 55	0.001 45
b_{116}	−0.000 136	1.78	0.001 59	0.000 568	−0.001 43	15.2	0.005 98
b_{117}	0.000 229	0.000 837	0.000 677	0.000 151	0.0015	0.005 99	17.7
b_{122}	9.27	−0.0013	8.05	−0.726	−0.002 35	−0.000 538	0.001 28
b_{123}	0.001 66	16.1	0.0305	0.004 49	6.35	6.25	0.004 07
b_{124}	0.001 11	−1.42	0.007 49	0.0126	−0.001 72	−10.6	−0.0126
b_{125}	−0.005 54	0.0183	6.34	−0.0112	0.066	−0.035	19.7
b_{126}	3.56	−0.0183	6.21	−10.7	−0.0105	0.0428	0.0112
b_{127}	−0.002 41	0.0166	0.006 64	−0.0206	19.7	−0.009 67	0.0844
b_{133}	12.4	−0.002 28	0.004 13	4.64	−0.0071	−0.004 43	7.58
b_{134}	0.0048	−0.003 18	9.32	0.0189	−0.002 45	−0.006 39	0.0143
b_{135}	−0.004 24	6.36	0.0363	−0.0112	0.0974	−0.521	−0.0177
b_{136}	0.003 84	6.21	−0.0216	−0.0119	−0.542	16.6	0.004 81
b_{137}	0.002 52	−0.000 397	15.2	0.0135	−0.003 52	0.0119	0.0899
b_{144}	9.27	−0.000 675	−0.003 47	−9.85	−0.000 607	−0.003 13	0.000 111
b_{145}	0.002 42	0.000 722	−0.002 71	0.0218	15.9	0.006 73	0.005 72
b_{146}	0.005 68	−10.7	−0.007 69	−0.0249	0.006 69	−18.9	−0.0288
b_{147}	0.003 89	−0.0118	0.000 269	0.0215	0.006 07	−0.0287	22
b_{155}	14.7	−0.001 57	0.004 55	7.94	−0.0126	−0.005 52	0.0137
b_{156}	−0.000 589	−0.0212	−0.538	0.006 03	−0.045	0.061	−6.76
b_{157}	−0.002 01	19.7	−0.008 78	0.002 91	0.101	−6.72	0.0823
b_{166}	15.2	0.001 19	8.28	−9.47	0.007 85	−0.0112	−0.000 778
b_{167}	0.0142	−0.008 18	0.002 34	−0.0304	−6.77	0.0373	−0.058
b_{177}	17.7	−0.000 84	0.000 344	11	−0.0171	−0.004 68	0.009 94
b_{222}	−0.000 543	19.8	−0.002 15	−0.000 668	1.02	2.67	−0.000 844
b_{223}	8.07	−0.005 99	52.3	10.5	−0.0115	0.001 51	8.95
b_{224}	−0.706	−0.001 92	10.5	16.8	−0.008 94	−0.004 12	0.002 04
b_{225}	−0.002 13	3.05	−0.0114	−0.008 62	55.9	−1.93	0.007 35
b_{226}	−0.000 816	8.01	0.001 71	−0.004 11	−1.93	48.1	0.005 94
b_{227}	0.001 19	−0.001 99	8.95	0.001 66	0.007 04	0.0057	61.5
b_{233}	−0.002 06	52.3	−0.0137	−0.002 36	12.6	−3.77	−0.0189
b_{234}	−0.0146	21	−0.0601	−0.0471	12.4	−15.8	−0.0445
b_{235}	6.35	−0.0508	25.2	12.4	−0.0911	0.0102	69.1
b_{236}	6.22	0.0293	−7.55	−15.8	0.000 848	0.0689	−15.5
b_{237}	−0.0167	17.9	−0.0971	−0.0539	69.1	−15.4	−0.152
b_{244}	−0.000 209	16.8	−0.002 72	0.007 93	−0.003 88	7.08	0.008 64
b_{245}	−0.0122	−0.0247	12.4	−0.0319	−0.0167	−0.006 25	40.5
b_{246}	−10.7	−0.006 05	−15.9	14.1	−0.0167	0.0161	0.0135
b_{247}	−0.0189	−0.0024	−0.0373	0.000 754	40.5	0.018	−0.0335
b_{255}	0	55.9	0	0.0076	0.0398	−24	−0.0056
b_{256}	−0.002 05	−3.84	−0.0162	−0.0172	−48	10.3	−0.0348
b_{257}	19.7	−0.006 18	69.2	40.5	−0.0827	−0.0124	−0.0256
b_{266}	0.001 76	48.1	0.0702	0.023	5.17	35.9	0.118
b_{267}	0.008 44	0.0355	−15.5	0.0103	−0.0338	0.186	−77.7
b_{277}	0.000 185	61.5	−0.023	0.001 07	0.0468	−38.9	0.0243
b_{333}	0.001 15	−0.004 14	61.9	0.0048	−0.005 11	−0.007 29	0.0212
b_{334}	4.66	−0.001 64	0.0138	55.9	0	−0.0191	26.5
b_{335}	−0.007 93	12.6	−0.0143	0	152	−2.4	0.0268
b_{336}	−0.004 44	−3.81	−0.0208	−0.0186	−2.39	64.3	−0.0475
b_{337}	7.6	−0.0192	0.0608	26.5	0.0279	−0.047	176
b_{344}	−0.003 85	−0.002 06	55.9	−0.0123	−0.008 29	−0.009 12	−0.0108
b_{345}	−0.0124	12.5	−0.005 19	−0.0352	−0.017	−28.7	−0.0256
b_{346}	0.002 01	−15.9	−0.0154	0.002 21	−28.7	13.7	−0.067
b_{347}	−0.014	−0.024	53	−0.0387	−0.009 48	−0.0795	−0.147
b_{355}	0.009 56	−0.001 26	152	0.005 76	−0.008 92	−0.042	−7.13
b_{356}	−0.529	−0.002 54	−4.78	−28.7	−0.0461	0.116	−44.9
b_{357}	−0.0227	69.2	0.006 98	−0.0347	−14.3	−44.9	0.0142
b_{366}	8.29	0.0686	64.4	6.92	0.0842	−0.1	29.9
b_{367}	0.0231	−15.5	−0.0675	−0.07	−44.9	59.8	−0.0377
b_{377}	0.003 38	−0.0235	176	−0.046	0.0735	−0.0457	0.125
b_{444}	−3.29	0.002 82	−0.004 18	22.9	−0.003 46	−0.001 54	0.004 13
b_{445}	−0.000 242	−0.002 63	−0.008 63	−0.01	67.6	−0.0102	0.0115
b_{446}	−0.003 39	7.05	−0.0088	−0.005 74	−0.009 78	56.6	−0.0187
b_{447}	0	0.008 91	−0.0111	0.0133	0.0115	−0.0188	75.2
b_{455}	7.96	0.006 57	0.001 21	67.6	0.0512	0.0131	0.0261
b_{456}	0.0158	−0.003 53	−28.7	−0.009 72	0.0416	0.0927	−89.9
b_{457}	−0.006 49	40.5	−0.0109	0.008 95	0.0352	−89.9	0.0487
b_{466}	−9.47	0.0233	6.92	56.6	0.0605	−0.12	−0.0198
b_{467}	−0.0245	0.0168	−0.0697	−0.0297	−89.9	−0.0576	−0.005 41
b_{477}	11	0.000 208	−0.0465	75.2	0.04	−0.0123	−0.0864
b_{555}	−0.004 19	0.0145	−0.003 16	0.0172	133	−0.0259	0.037
b_{556}	−0.005 27	−24	−0.0436	0.0127	−0.0797	116	−0.171
b_{557}	0.0127	−0.01	−7.14	0.027	0.115	−0.171	333
b_{566}	0.009 11	5.17	0.0853	0.0608	116	−18.9	0.0809
b_{567}	−6.76	−0.0154	−44.9	−89.9	−0.323	0.127	0.065
b_{577}	−0.0201	0.054	0.0681	0.0404	333	0.009 68	0.0256
b_{666}	−0.004 26	12	−0.0328	−0.0392	−6.29	96.8	−0.0741
b_{667}	−0.001 24	0.119	29.9	−0.0198	0.0806	−0.223	151
b_{677}	−0.004 42	−38.9	−0.0458	−0.012	0.0149	151	−0.457
b_{777}	0.002 68	0.008 51	0.0411	−0.0296	0.007 24	−0.153	273

Table E-11: Coeff. for a **square membrane** (as Table 3-19). Modes: 1,2,4,5,7,9,11. Error: $\epsilon_{\max} < 7.5\%$, $\epsilon_{\text{mean}} < 0.87\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
b_{111}	3.65	0	0	−0.0807	0	−0.000 131	−0.0466
b_{112}	0	11.6	0	−0.000 267	1.17	−1.17	−0.000 234
b_{113}	0	0	15.2	−0.000 269	0	0	−0.001 12
b_{114}	−0.242	0	−0.000 418	16.4	−0.000 231	−0.000 154	−0.635
b_{115}	0	1.15	0.000 103	−0.000 291	12.9	−1.27	0.001 07
b_{116}	0	−1.13	−0.000 234	−0.000 123	−1.27	31.8	0.002 11
b_{117}	−0.144	−0.000 263	−0.001 17	−0.634	0.000 812	0.001 59	27.4
b_{122}	11.6	−0.000 387	10.4	2.87	−0.000 952	−0.0015	0.367
b_{123}	−0.0597	20.7	−0.303	−0.0311	−13.7	−8.35	−0.109
b_{124}	−0.0514	5.63	−0.0362	−0.385	4.47	7.54	0.179
b_{125}	2.29	−0.057	−14.1	4.49	−0.193	−0.0697	−18
b_{126}	−2.32	−0.126	−8.16	8	−0.0441	−0.494	−9.19
b_{127}	0.0561	0.791	0.0117	0.0806	−18.2	−8.97	−0.397
b_{133}	15.2	−0.000 448	0.002 01	1.2	0.001 78	−0.003 27	8.13
b_{134}	−0.019	0.001 78	2.29	−0.186	0.000 313	0.006 95	−0.0058
b_{135}	−0.0147	−14	−0.121	−0.005 15	10.3	−9.72	−0.16
b_{136}	−0.0233	−8.18	−0.14	−0.0121	−9.85	−3.8	−0.0321
b_{137}	0.0116	0.001 07	16.5	−0.007 81	0.007 65	−0.006 99	−0.232
b_{144}	16.4	0	0	2.26	−0.001 41	0.004 65	−19.8
b_{145}	−0.0133	4.48	0.003 81	−0.175	3.1	−8.76	0.0714
b_{146}	−0.0633	8.01	0.005 51	−0.26	−8.47	19.2	0.227
b_{147}	−1.16	0.002 15	−0.005 16	−39.3	−0.000 571	0.0115	−0.279
b_{155}	12.9	−0.003 23	5.31	1.64	−0.001 01	−0.001 47	2.64
b_{156}	−2.53	−0.0488	−9.79	−8.34	−0.148	−0.267	−17.9
b_{157}	0.0263	−18.2	−0.0387	0.0259	5.39	−17.6	−0.312
b_{166}	31.8	0.0107	−1.71	9.94	0.0026	−0.0312	0.307
b_{167}	0.061	−9.15	0.000 304	0.189	−17.8	1	−0.458
b_{177}	27.4	0	0.0117	0.2	0.0051	−0.0057	0.11
b_{222}	−0.000 165	21.9	0	0.000 103	−0.117	−1.95	0.000 453
b_{223}	10.5	0.000 308	58.2	2.24	0.002 78	0.0003	8.15
b_{224}	2.91	0	2.24	58.7	−0.003 07	0.005 54	−6.02
b_{225}	−0.000 944	−0.34	0.002 68	−0.002 86	51	−6.71	0.007 85
b_{226}	−0.001 46	−5.86	−0.000 207	0.004 51	−6.71	79.3	−0.000 829
b_{227}	0.307	0.001 35	8.17	−6.03	0.007 68	−0.001 78	69.9
b_{233}	−0.000 749	58.2	−0.0025	−0.001 13	−28.8	−2.03	−0.0032
b_{234}	−0.0416	4.4	−0.115	−0.19	−7.88	37.6	0.0978
b_{235}	−14.1	0.0428	−57.7	−7.9	−0.224	0.0697	−64.9
b_{236}	−8.21	−0.0898	−4.19	37.7	0.144	−0.495	11.6
b_{237}	−0.008 33	16.4	−0.0144	0.0216	−64.9	11.6	−0.316
b_{244}	0	58.7	0.003 45	0.002 42	34.5	21.1	0.001 02
b_{245}	4.56	−0.0935	−7.91	68.8	−0.267	−0.195	−26.9
b_{246}	8.12	−0.169	37.7	42	−0.116	−0.904	−58.8
b_{247}	0.0902	−12	0.0513	0.398	−26.9	−58.7	−0.63
b_{255}	−0.003 67	51	−0.008 05	−0.006 74	−20.2	10.4	−0.0024
b_{256}	−0.000 602	−13.5	0.0222	−0.0165	20.7	46	0.0198
b_{257}	−18.3	0.0616	−65	−26.9	0.146	0.128	−15.4
b_{266}	0.0114	79.4	0.009 16	0.0192	23.2	4.87	−0.0265
b_{267}	−9.2	0.121	11.6	−59.1	0.0295	0.535	72.6
b_{277}	0.001 43	70	−0.006 57	−0.02	−7.5	36.4	−0.0217
b_{333}	0.000 617	−0.0005	58.5	−0.002 02	0.003 16	0.000 402	0.004 16
b_{334}	1.21	0	−0.006 48	67	−0.003 99	0.003 73	4.26
b_{335}	0.001 61	−28.9	0.009 64	−0.004 11	96.5	3.74	0.011
b_{336}	−0.0036	−2.01	−0.000 215	0.002 01	3.72	145	−0.004 39
b_{337}	8.21	−0.002 74	0.0113	4.26	0.0106	−0.004 83	156
b_{344}	0	0.002 94	67	0.001 08	0.005 33	−0.000 892	−0.007 72
b_{345}	0.009 29	−7.9	−0.0623	−0.089	10.9	−16.8	−0.0892
b_{346}	−0.0278	37.8	−0.0846	−0.116	−16.8	2	0.0116
b_{347}	0.006 47	0.0126	8.55	0.122	0.001 04	−0.0017	−0.208
b_{355}	5.3	−0.008 47	96.5	5.51	0.028	−0.0255	71.2
b_{356}	−9.82	0.0311	7.38	−16.9	−0.134	−0.235	−19.5
b_{357}	−0.0299	−65.1	0.005 07	0.006 18	143	−19.5	−0.193
b_{366}	−1.73	0.0104	145	1.16	−0.0114	−0.097	−12.7
b_{367}	−0.0406	11.6	0.0688	−0.0194	−19.6	−25.1	−0.472
b_{377}	0.0107	−0.007 93	156	0.0176	0.008 23	−0.0476	0.137
b_{444}	0.767	0.000 402	0.000 369	81.4	−0.003 72	0.0116	−7.64
b_{445}	−0.001 48	34.5	0.005 03	−0.0106	112	41	0.0106
b_{446}	0.002 54	21.2	−0.003 15	0.0327	40.9	192	−0.0431
b_{447}	−20.1	0.003 05	−0.007 74	−22.9	0.0118	−0.0465	253
b_{455}	1.66	−0.008 14	5.52	112	−0.008 63	−0.0149	−9.73
b_{456}	−8.43	−0.114	−16.9	81.7	−0.224	−0.459	−80.8
b_{457}	0.0719	−26.9	−0.004 88	0.178	−19.3	−80.8	−0.367
b_{466}	9.98	0.0206	1.16	192	0.0156	0.393	−103
b_{467}	0.195	−59.1	−0.002 07	0.455	−80.9	−205	−0.59
b_{477}	0.215	−0.0217	0.0231	252	−0.043	0.003 34	0.049
b_{555}	−0.000 449	−6.75	0.009 86	−0.002 93	76.1	−9.91	0.0205
b_{556}	0.001 39	10.4	−0.0263	−0.0158	−29.7	158	−0.0479
b_{557}	2.57	−0.003 12	71.3	−9.74	0.0581	−0.0474	240
b_{566}	0.005 79	23.2	−0.0138	0.0161	158	−0.405	−0.0349
b_{567}	−17.9	0.118	−19.6	−81.1	−0.0376	0.243	−0.642
b_{577}	0.004 84	−7.5	0.006 27	−0.0436	240	−0.186	−0.007 17
b_{666}	−0.0145	1.63	−0.0311	0.139	−0.136	357	−0.105
b_{667}	0.147	−0.0237	−12.7	−103	−0.0374	−0.317	406
b_{677}	−0.005 76	36.6	−0.0549	0.001 93	−0.251	406	0.145
b_{777}	−0.0396	−0.006 22	0.0474	0.0102	−0.001 17	0.0459	295

Table E-12: Coeff. for **A. Keşkekler's membrane**. Modes: 1,2,4,6,7,9,15. Error: $\epsilon_{\max} < 15\%$, $\epsilon_{\text{mean}} < 1\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
b_{111}	2.55	0	0.313	1.34	0.000 204	0.000 435	−0.529
b_{112}	0	8.57	−0.000 307	−0.000 596	−0.426	−4.05	0.000 196
b_{113}	0.942	−0.000 283	9.79	1.07	−0.001 11	0.000 244	−0.0467
b_{114}	4.02	−0.000 585	1.07	10.8	0.001 24	−0.000 745	2.96
b_{115}	0.000 58	−0.434	−0.001 09	0.001 24	12	0.71	0.000 527
b_{116}	0.001 28	−4.05	0.000 289	−0.000 762	0.711	15.9	0.0028
b_{117}	−1.61	0.000 168	−0.0469	2.96	0.000 479	0.0028	14.2
b_{122}	8.61	0	−5.82	3.82	0.001 54	0.002 08	−2.56
b_{123}	−0.143	−11.8	0.349	0.0771	−13.8	7.28	0.0707
b_{124}	0.001 29	7.69	0.0838	0.167	0.621	−14.5	0.08
b_{125}	−0.826	0.174	−13.7	0.625	0.434	0.0744	2.17
b_{126}	−8.12	0.0522	7.31	−14.5	0.0669	0.26	0.184
b_{127}	−0.0667	−5.07	0.0861	0.0629	2.17	0.158	0.361
b_{133}	9.88	0.000 849	1.46	1.37	−0.000 543	0.002 17	−7.65
b_{134}	2.13	0.000 409	2.83	6.92	0.002 48	−0.000 509	4.82
b_{135}	−0.0277	−13.7	0.22	−0.000 755	1.95	6.94	−0.0145
b_{136}	−0.001 52	7.33	0.085	0.019	6.94	−9.79	−0.006 23
b_{137}	−0.0863	−0.000 881	−15.3	4.82	0.0112	−0.003 04	10.5
b_{144}	10.8	−0.000 906	3.54	17.4	0.001 68	−0.000 612	8.08
b_{145}	0.0254	0.634	0.0135	0.133	−1.84	2.25	−0.0204
b_{146}	0.0829	−14.4	0.0383	0.167	2.25	34	−0.006 71
b_{147}	5.92	−0.005 01	4.86	16.3	0.003 22	0.0132	28.8
b_{155}	12.1	−0.000 487	1.18	−1.01	−0.001 61	0.005 79	−11.1
b_{156}	1.45	0.02	7.02	2.27	0.197	0.293	−0.336
b_{157}	−0.032	2.16	−0.000 708	−0.0264	−22.1	−0.336	0.136
b_{166}	16.1	0.002 54	−4.79	16.8	0.006 14	−0.0112	2.66
b_{167}	−0.033	0.161	−0.001 44	−0.0053	−0.335	5.38	0.144
b_{177}	14.4	−0.006 84	5.31	14.3	0.002 16	0.0112	3.13
b_{222}	0	18.7	−0.000 908	−0.001 43	1.94	−6.99	0.000 362
b_{223}	−5.73	−0.002 72	45.2	6.43	−0.004 45	−0.001 77	6.38
b_{224}	3.83	−0.004 33	6.42	16	0.001 59	−0.001 27	3.57
b_{225}	0.001 46	5.81	−0.004 34	0.001 56	47.7	4.2	−0.009 25
b_{226}	0.001 94	−21	−0.001 57	−0.001 27	4.21	53	0.0126
b_{227}	−2.48	0.000 901	6.38	3.55	−0.009 39	0.0125	22.2
b_{233}	0.000 75	45.2	−0.004 85	−0.004 84	31.9	−5.51	−0.004 98
b_{234}	0.004 52	12.9	−0.005 23	−0.0825	20.3	7.73	−0.0346
b_{235}	−13.5	−0.224	64	20.3	−0.399	−0.0729	15.6
b_{236}	7.3	0.0157	−10.9	7.74	−0.0876	−0.285	−9.19
b_{237}	−0.0112	12.8	−0.0207	−0.0566	15.6	−9.18	−0.158
b_{244}	−0.000 933	16	−0.002 87	−0.003 74	−3.94	−14.1	−0.007 15
b_{245}	0.605	0.008 98	20.3	−7.81	0.0907	0.002 22	−2.75
b_{246}	−14.3	−0.0382	7.73	−28.1	0.002 47	0.15	−11.7
b_{247}	0.0374	7.29	0.0448	0.237	−2.76	−11.8	0.376
b_{255}	−0.000 489	47.8	0.004 69	−0.001 45	0.0154	10.7	−0.012
b_{256}	−0.004 83	8.51	−0.004 51	−0.0111	21.5	5.05	0.000 509
b_{257}	2.13	−0.0207	15.6	−2.74	0.007 64	0.006 18	−12.7
b_{266}	0.002 43	53.3	−0.0148	0.004 74	2.42	−66.2	−0.018
b_{267}	−0.108	−0.0164	−9.17	−11.7	0.0066	0.0415	−29.3
b_{277}	−0.006 64	22.3	−0.004 88	−0.0109	−6.38	−14.7	−0.005 34
b_{333}	0.476	−0.001 44	44.9	4.06	−0.005 68	−0.001 14	0.176
b_{334}	1.35	−0.004 73	12.2	45.4	−0.000 223	−0.0106	8.8
b_{335}	−0.000 782	31.8	−0.0168	−0.000 288	121	11.2	−0.006 72
b_{336}	0.002 11	−5.46	−0.002 99	−0.0106	11.2	54.8	−0.008 19
b_{337}	−7.59	−0.005 04	0.53	8.78	−0.007 05	−0.008 23	45.1
b_{344}	3.5	−0.002 79	45.4	0.37	−0.005 26	−0.003 48	5.32
b_{345}	−0.005 34	20.2	0.0276	−0.0768	5.01	58.3	−0.0532
b_{346}	−0.007 06	7.68	−0.006 18	−0.0376	58.3	12.7	0.002 49
b_{347}	4.81	−0.000 293	17.6	10.5	−0.0412	0.0192	4.01
b_{355}	1.15	0.004 91	121	2.55	0.000 372	−0.002	−6.49
b_{356}	6.9	−0.0452	22.4	58.3	−0.145	−0.213	−14
b_{357}	−0.000 73	15.5	0.003 02	−0.0546	−13	−14	−0.085
b_{366}	−4.85	−0.0145	54.9	6.4	−0.0298	0.009 73	11.9
b_{367}	−0.005 78	−9.12	−0.003 29	0.001 84	−14	23.8	−0.0382
b_{377}	5.24	−0.004 72	45.1	2.09	−0.003 47	0.0116	−2.47
b_{444}	5.72	−0.001 18	0.121	26.5	0.001 94	−0.001 52	13.7
b_{445}	0.001 57	−3.92	−0.005 16	0.005 24	57.3	−4.5	0.0103
b_{446}	−0.000 414	−14	−0.003 38	−0.005 93	−4.5	61.5	0.0181
b_{447}	8.05	−0.007 03	5.31	41.1	0.0101	0.0182	70.8
b_{455}	−1.05	−0.001 39	2.54	57.3	0.004 56	−0.003 35	−11.8
b_{456}	2.22	−0.007 29	58.3	−8.97	0.0408	0.0648	8.01
b_{457}	0.0154	−2.73	−0.0417	0.0565	−23.6	8.01	0.0482
b_{466}	16.5	0.005 02	6.39	61.5	−0.007 38	−0.0574	24.7
b_{467}	0.0212	−11.6	0.0228	0.0718	8.01	49.5	0.129
b_{477}	14.1	−0.0106	2.08	70.8	−0.006 54	0.0288	77.4
b_{555}	−0.000 543	0.0125	0.000 385	0.001 93	116	2.28	−0.006 66
b_{556}	0.005 66	10.7	−0.001 35	−0.003 52	6.83	105	−0.0293
b_{557}	−10.9	−0.012	−6.49	−11.8	−0.0234	−0.0292	61.7
b_{566}	0.006 17	2.45	−0.0296	−0.007 28	105	42.2	−0.017
b_{567}	−0.308	−0.001 15	−14	8.01	−0.0505	−0.0185	−4.96
b_{577}	0.001 92	−6.34	−0.003 34	−0.006 72	61.8	−2.49	0.0524
b_{666}	−0.002 65	−21.9	0.0039	−0.0176	14.1	99.4	0.0251
b_{667}	2.89	−0.0179	11.9	24.7	−0.0172	0.077	67.3
b_{677}	0.0108	−14.5	0.0118	0.0285	−2.49	67.3	0.0274
b_{777}	1.13	−0.000 123	−0.814	25.8	0.0211	0.006 82	81.4

Table E-13: Coeff. for an **elliptic membrane**. Modes: 1-7. Error: $\epsilon_{\max} < 0.8\%$, $\epsilon_{\text{mean}} < 0.09\%$

	Eq. 1	Eq. 2	Eq. 3	Eq. 4	Eq. 5	Eq. 6	Eq. 7
b_{111}	2.46	0	−0.586	0	−0.000 189	0	0.0028
b_{112}	−0.000 163	6.57	0	0.000 146	2	−0.000 218	0.000 989
b_{113}	−1.76	0	6.48	−0.001 24	0.001 88	0.001 01	−2.11
b_{114}	−0.000 144	0.000 169	−0.001 21	8.69	0.000 954	−0.0025	−0.0061
b_{115}	−0.000 574	2	0.001 92	0.000 976	6.45	−0.001 12	−0.008 07
b_{116}	−0.000 282	−0.000 27	0.001 03	−0.002 46	−0.001 16	9.68	0.004 49
b_{117}	0.008 18	0.000 99	−2.11	−0.006 09	−0.008 14	0.004 46	6.78
b_{122}	6.57	0.000 125	4.44	−0.000 204	0.000 97	0.001 27	−2.27
b_{123}	−0.004 11	8.88	−0.0197	−0.0022	−8.17	0.002 49	0.008 87
b_{124}	−0.002 49	−0.006 85	−0.003 21	−0.0208	−0.008 99	−12.8	−0.006 38
b_{125}	4	0.004 18	−8.18	−0.004 17	0.005 89	−0.000 361	−7.82
b_{126}	0.004 21	0.0119	0.006 75	−12.8	0.003 63	−0.0357	−0.0609
b_{127}	0.002 66	−4.53	0.0193	−0.008 09	−7.83	−0.0572	−0.0647
b_{133}	6.48	−0.001 32	1.43	0.003 24	−0.009 94	0.001 96	3.63
b_{134}	0.000 751	−0.002 42	−0.000 275	−11.5	0.000 366	0.005 71	0.0535
b_{135}	0.005 26	−8.18	−0.0124	0	2.38	0.0121	0.0446
b_{136}	0.004 29	0.002 62	0.0121	0.0057	0.0122	2.4	−0.036
b_{137}	−4.23	0.0171	7.27	0.0501	0.0337	−0.0337	1.97
b_{144}	8.7	−0.001 88	−5.77	0.005 55	−0.003 84	−0.003 23	1.73
b_{145}	0.003 38	−0.009 27	−0.000 303	0.000 859	−0.004 52	−13.3	−0.0286
b_{146}	−0.001 72	−12.8	0.005 84	0.0192	−13.3	−0.0217	−0.008 32
b_{147}	−0.0149	−0.007 81	0.0559	3.44	−0.0277	−0.005 54	−0.132
b_{155}	6.45	0.007 75	1.19	0.001 62	0.005 35	−0.0106	0.925
b_{156}	−0.004 02	0.003 91	0.0123	−13.3	−0.0112	0.0243	−0.0108
b_{157}	−0.0156	−7.83	0.0372	−0.0281	1.84	−0.0116	0.0267
b_{166}	9.68	−0.000 366	1.21	0.007 63	0.004 93	0.0131	−7.49
b_{167}	0.006 13	−0.0574	−0.0379	−0.006 05	−0.0119	−15	0.0398
b_{177}	6.78	−0.0257	0.991	−0.0619	0.006 26	0.0147	0.191
b_{222}	0	6.93	0	−0.000 58	0.137	0.002 55	0.004 56
b_{223}	4.44	−0.000 351	17.6	0	−0.007 08	0.006 86	−1.1
b_{224}	−0.000 312	−0.0017	0	12.1	0.003 75	−0.000 637	0.0209
b_{225}	0.000 943	0.412	−0.007 12	0.0038	18	0.009 84	−0.005 54
b_{226}	0.001 06	0.0074	0.006 76	−0.000 685	0.0099	21.7	−0.0124
b_{227}	−2.27	0.0137	−1.1	0.0208	−0.005 52	−0.0124	19.3
b_{233}	−0.001 32	17.6	0.0108	−0.006 45	−11.3	0.005 34	0.009 79
b_{234}	−0.003 31	−0.001 54	−0.0165	−0.004 72	−0.0126	−11.8	0.0313
b_{235}	−8.18	0.000 151	−22.6	−0.0139	0.0137	0.0162	−22.1
b_{236}	0.006 16	0.0192	0.0214	−11.8	0.0129	−0.0248	−0.134
b_{237}	0.0151	−2.2	0.007 96	0.0287	−22.1	−0.129	−0.192
b_{244}	−0.002 18	12.1	0	0.001 96	8.47	−0.006 34	−0.000 944
b_{245}	−0.004 74	0.006 87	−0.0138	16.9	0.0167	−0.0117	−0.017
b_{246}	−12.8	0.005 92	−11.9	0.006 44	−0.0082	−0.0624	17.7
b_{247}	−0.006 74	0.0403	0.0321	−0.006 39	−0.0173	17.7	−0.0219
b_{255}	0.007 71	18	0.0242	0.0102	−3.32	0.0109	−0.0459
b_{256}	−0.000 223	0.0175	0.0166	−0.0119	0.0336	14.3	0.0436
b_{257}	−7.83	−0.006 95	−22.1	−0.0172	−0.11	0.0433	−6.62
b_{266}	−0.000 572	21.7	−0.001 15	−0.0168	7.17	0.0328	0.0384
b_{267}	−0.0632	−0.0225	−0.132	17.7	0.0459	0.0639	0.174
b_{277}	−0.0259	19.3	−0.0836	−0.007 47	−3.33	0.0799	0.006 93
b_{333}	0.476	0.003 54	14.9	0.001 58	−0.003 01	0.001 47	0.578
b_{334}	0.002 94	−0.006 17	0.005 28	15.9	−0.003 24	−0.000 745	−0.001 26
b_{335}	−0.0103	−11.3	−0.009 32	−0.003 11	35.5	−0.008 54	−0.009 65
b_{336}	0.0021	0.005 29	0.004 73	−0.000 859	−0.008 26	26.3	−0.0151
b_{337}	3.64	0.009 98	1.73	−0.001 29	−0.009 47	−0.015	37
b_{344}	−5.77	0	15.9	−0.0418	0.0113	0.0161	−10.3
b_{345}	−0.000 73	−0.0135	−0.004 84	0.025	−0.007 78	15.5	0.0257
b_{346}	0.005 41	−11.9	0.001 26	0.0382	15.5	−0.0513	−0.0554
b_{347}	0.0487	0.0289	−0.001 42	−20.7	0.0237	−0.0553	0.16
b_{355}	1.19	0.0247	35.5	−0.001 58	−0.006 21	0.000 133	22.5
b_{356}	0.0131	0.011	−0.0196	15.5	0.008 43	−0.0505	0.132
b_{357}	0.0409	−22.1	0.005 09	0.024	45.1	0.136	0.421
b_{366}	1.21	−0.000 984	26.3	−0.0213	−0.0295	0.0718	−9.13
b_{367}	−0.0335	−0.13	−0.0277	−0.0559	0.136	−18.3	0.158
b_{377}	0.991	−0.0838	37	0.0819	0.183	0.0745	7.39
b_{444}	0.002 37	0.000 559	−0.014	26.4	0.004 37	0.0081	−0.0244
b_{445}	−0.003 67	8.47	0.011	0.0133	17.7	−0.03	−0.0435
b_{446}	−0.003 28	−0.005 93	0.0164	0.0243	−0.0298	61.2	0.0478
b_{447}	1.73	−0.001 13	−10.3	−0.0733	−0.0435	0.0479	19.4
b_{455}	0.001 92	0.0102	−0.001 58	17.7	0.0185	−0.0328	0.0372
b_{456}	−13.3	−0.008 04	15.5	−0.0665	−0.057	0.0608	15.8
b_{457}	−0.0286	−0.0193	0.0228	−0.0854	0.0724	15.8	0.0521
b_{466}	0.006 69	−0.0169	−0.0211	61.2	0.0258	−0.041	0.139
b_{467}	−0.0104	17.7	−0.057	0.0991	15.8	0.271	0.0768
b_{477}	−0.062	−0.007 76	0.0822	19.4	0.023	0.0344	−0.0223
b_{555}	0.001 77	−1.11	−0.002 29	0.006 08	26.9	0.0114	0.008 19
b_{556}	−0.0108	0.011	0.000 592	−0.0329	0.0354	36.2	0.0937
b_{557}	0.922	−0.0462	22.6	0.0371	0.0241	0.0938	63.5
b_{566}	0.005 19	7.17	−0.0295	0.0258	36.2	0.007 07	−0.0144
b_{567}	−0.0109	0.045	0.135	15.8	0.192	−0.0222	0.312
b_{577}	0.006 58	−3.32	0.183	0.0236	63.5	0.161	0.215
b_{666}	0.0044	0.0108	0.0239	−0.0131	0.002 44	47.9	−0.0118
b_{667}	−7.5	0.0378	−9.13	0.138	−0.0143	−0.0354	41.1
b_{677}	0.0152	0.0806	0.0745	0.0342	0.161	41.1	−0.0726
b_{777}	0.0627	0.002 36	2.47	−0.007 73	0.0724	−0.0245	45.9

Appendix F

Code

In this appendix, a selection of the written code is shown. Some scripts written automatically by Matlab are shown, like 'duffing.for' and 'fenergy_mode.m'. Those scripts automatically written for a given set of coefficients (of the equations of motion), and in practice are very large. To provide insight in their structure, they are shown here for a ROM of 2 modes. Scripts that write Fortran scripts are not shown, as they are quite incomprehensible.

The commands for the API can be found from the 'Comsol API Reference Manual'. However, it is very difficult to program a model from scratch this way. Therefore, it is recommended to build a model in the Comsol GUI and save it as m-file. From this file, one can exactly learn how the commands should be used.

F-1 Matlab-Comsol interface: STEP method

F-1-1 Element settings

```
1  function [a_coefname,b_coefname,a_coef_un,b_coef_un,Km_value_un, MD]...
2   = fsnaar_model_out_of_plane(n_eig_solve,ind_eig,MP)
3
4  % Snaar Step method
5  % Vincent Bos
6  % 26-3-2019
7
8  %{
9  @I/O@ : this line determines type of analysis: include in-plane or not
10 @ST@  : this line is structure dependent
11 %}
12
13 %% 1. INITIALISE COMSOL INTERFACE
14 import com.comsol.model.*
15 import com.comsol.model.util.*
16 % ModelUtil.tags                % geeft alle open modellen weer
```

```
17 ModelUtil.clear;
18 ModelUtil.showProgress(true);    % toon progress
19
20 fstruc2var(MP);
21
22 % SETTINGS
23 disp_factor = 2;
24 fact_disp_in= 1; % hoeveel in-plane kleiner moet zijn dan out-of-plane
25 n_ele       = 100;
26 n_eig       = length(ind_eig);
27 ind_in      = [];
28
29 %% 2. COMSOL INTERFACE
30
31 model = mphopen('snaar');    % @ST@ open model
32
33 % SET PHYSICAL AND GEOMETRICAL MODEL PROPERTIES whole section: @ST@
34 model.param.set('L',    [num2str(Length)    '[m]'], 'length');
35 model.param.set('As',   [num2str(As)        '[m]'], 'diameter string');
36 model.param.set('Emod', [num2str(Emod)     '[Pa]'], 'Young''s modulus');
37 model.param.set('nu',   [num2str(nu)              ], 'Poisson''s ratio');
38 model.param.set('rho',  [num2str(rho)  '[kg/m^3]'], 'density');
39 model.param.set('sig0', [num2str(sig0)     '[Pa]'], 'prestress');
40
41 % settings eigenvalue analysis
42 model.param.set('n_ele',[num2str(n_ele)          ], 'number of elements')
       ;  % @ST@
43 model.sol('sol1').feature('e1').set('neigs', n_eig_solve);     % @I/O@ //
       OUT-OF-PLANE \\ # eigenmodes
44
45 model.sol('sol1').feature('e1').set('shift', '1[Hz]');         % search
       around freq
46 model.sol('sol1').feature('e1').set('eigvfunscale', 'mass');   % scale
       eigenvectors to make mass matrix identity
47
48 % run eigenvalue analysis
49 model.sol('sol1').runAll;
50
51 % extract solution
52 [MS] = fSTEP_sol(model,ind_eig); fstruc2var(MS);
53
54 % EXTRACT MESH INFO
55 [MM] = fSTEP_mesh(model); fstruc2var(MM);
56
57 % ~~ structure settings ~~ %
58 study_nr            = 2;              % @ST@ index node nonlinear solver
59 physics_name        = 'truss';        % @ST@ name of physics
60 arg_coord           = 'x,y';          % @ST@ coordinates for interp
       functions (string)
61 n_arg_coord         = 2;              % @ST@ # of coordinates: 2D: 2, 3D
       : 3
62 index_out_of_plane  = 1;              % @ST@ 0-based index dof-dir with
       largest amplitude
```

```
63 index_u              = [0  1]';          % @ST@ 0-based index of dof-dir
       related to displacements
64 dir_ind_v            = [1  2]';          % @ST@ 1-based index of dof-dir
       eigenvectors to be saved
65 n_f                  = [   1];           % @ST@ @I/O@ 0-based vector met
       richtingen die opgelegd worden: [  1]: LEAVE IN-PLANE FREE
66 dir_fix              = dir_name(n_f+1); % names of applied fields
67 dispname             = dir_name{2};     % @ST@ hoofdrichting, voornaamste
       verplaatsing modes <-- struct afhankelijk
68 dispname_in          = dir_name{1};     % @ST@ in-plane richting,
69
70 % select geometry
71 geom_index           = 1;               % @ST@ 1 for edge, 2 for domain
72 geom_selection       = [1];             % @ST@ selection of geometry
       entities
73
74 % INTERFACE & STEP METHOD
75 S_STEP_out_of_plane
76
77 % close Comsol model
78 ModelUtil.clear
79 end
```

F-1-2 Organizing block static condensation

```
1  %% STEP METHOD FOR ONLY OUT-OF-PLANE MODES
2  % Vincent Bos
3  % 24-5-2019
4
5  MT = fvar2struc(dir_fix,physics_name,index_out_of_plane,index_u,n_f,
       dispname,dispname_in,geom_index,geom_selection,dir_ind_v);
6
7  % extract FEM data
8  [MFC] = fSTEP_FEM_info_pre(model,MM,MS,MT);  fstruc2var(MFC);
9
10 mode_in      = [];
                                                          % @I/O@
11 U_in         = [];
                                                          % @I/O@
12 f_in         = [];
13
14 [MD,U_eig,Km,ind_eig,n_eig,n_eig_out] = fSTEP_FEM_in_out(model,MS,MM,MFC,
       MT,MP,U_in,mode_in,f_in); % ADD IN-PLANE EIGENVECTORS TO TOTAL SET (
       U_eig is different from the one in the stucture MS!)
15
16 %% 3. EXPORT EIGENVECTORS TO COMSOL
17
18 kmv          = diag(MFC.Km)' ./ MFC.Km(1,1) ;
19
20 U_vmax   = maxk(U_eig(dof_u,:),1,1,'ComparisonMethod','abs');
                    % <--- was struct dependent
```

```
21 qv      = disp_factor*thic ./ U_vmax ./ kmv;
                                          % scalar to lower displacements
   8* default
22 qm      = diag(qv);
23
24 % check coord for interpolation: remove y coord if curved string
25 if exist('ind_coord')
26     coord_disp_corr = coord_disp(:,ind_coord);
27 else
28     coord_disp_corr = coord_disp;
29 end
30
31 %  Storing eigenvectors
32 [filename,B] = fSTEP_csv(n_eig, U_eig, qm,n_node_disp,n_field_disp,
   dof_dir,n_f,coord_disp_corr);
33 csvwrite(filename,B)
34
35 % set interpolation functions and constraints
36 S_constr_interp_out
37
38 %% ~~                                                              ~~
    %
39 %% ************************** 4. STEP METHOD ***************************
    %
40
41 % coefficients: names & numbers
42 [SH] = fSTEP_h(n_eig); fstruc2var(SH)
43 disp([num2str(n_htot/8) ' sec computation time expected'])
44
45 % export h-matrix
46 S_h_comsol;
47
48 % Static analysis - run Comsol
49 F_lin_mod    = Km*qm*h;              % linear
50 tic, model.sol(solut_name).runAll; toc, % nonlinear analysis
51
52 F_tot       = mphgetu(model,'soltag',solut_name,'solnum',[1:1:n_htot],'
   type','reacf'); % read comsol model: NON-linear analysis
53 F_tot_mod   = U_eig' * F_tot;
54 F_nonlin    = F_tot_mod - F_lin_mod;
55
56 % determine coeff automatically
57 [A,B,a_coef,b_coef] = fSTEP_coef(SH,qv,F_nonlin);
58
59 %% 5. WRITE OUTPUT
60
61 n_dim_U          = n_eig*2 + 2;
62 MD.Emod          = Emod;
63 MD.U_vmax        = U_vmax;
64 MD.modal_mass    = modal_mass;
65
66 a_coef_un           = a_coef;
67 b_coef_un           = b_coef;
```

```
68 Km_value_un          = diag(Km);
69
70 path_overdracht = 'C:\Users\Vincent\Documents\2. School\2017-2018\
       Afstuderen\1. Matlab\BESTAND OVERDRACHT';
71 save([path_overdracht '\coefficients.mat'],...
72     'A','B','a_coefname','b_coefname','a_coef_un','b_coef_un','a_length',
           'b_length',...
73     'n_eig','Km_value_un','n_dim_U','ind_eig','MD','MP');
```

F-1-3 Create modal displacement matrix

```
1  function [SH] = fSTEP_h(n_eig)
2
3  % INPUT:    n_eig
4  % OUTPUT:   h, HT, n_htot,
5
6  % aantal coefficienten
7  L   = n_eig;     % # eigenmodes involved
8  i   = 1;
9
10 for j=1:L
11     for k=j:L
12         a_coefname(i,:)          = str2num([num2str(j) ' ' num2str(k)]);
13         a_coefnamevec(i,:)       ={[num2str(j) '  ' num2str(k)]};
14         a_coefnamematrix(j,k)    = str2num([num2str(j)     num2str(k)]);
15         i                        = i+1;
16     end
17 end
18
19 i   = 1;
20 for j=1:L
21     for k=j:L
22         for l=k:L
23             b_coefname(i,:)          = str2num([num2str(j) ' ' num2str(k)
                   ' ' num2str(l)]);
24             b_coefnamevec(i,:)       ={[num2str(j) '  ' num2str(k) '  '
                   num2str(l)]};
25             b_coefnamematrix(j,k,l) = str2num([num2str(j)     num2str(k)
                       num2str(l)]);
26             i                        = i+1;
27         end
28     end
29 end
30
31 a_length = (L*(L+1))/2;
32 b_length = (L*(L+1)*(L+2))/(3*2);
33 n_h1     = 2 *   L;
34
35 if (L > 1)
36     n_h2     = 3 *   factorial(L)/(factorial(2)*(factorial(L-2)));
37 else
38     n_h2     = 0;
39 end
```

```
40
41 if (L > 2)
42     n_h3    =   factorial(L)/(factorial(3)*(factorial(L-3)));
43 else
44     n_h3    = 0;
45 end
46
47 n_htot  = n_h1 + n_h2 + n_h3;
48
49 %% Assemble h-matrix: weights for prescribed displacement fields
50
51 % -- h1 --
52 h1_block    = [1   -1];
53 h1          = zeros(L,n_h1);
54 for i = 1 : n_h1/2
55     h1(i,[2*i-1 2*i])    = h1_block;
56     h1_name(i)     = {num2str(i)};
57     h1_name_ind(i) = 2*i-1;
58 end
59
60 H1  = containers.Map(h1_name,h1_name_ind);
61
62 % -- h2 --
63 h2_block_1  = [1   -1    1];
64 h2_block_2  = [1   -1   -1];
65 h2          = zeros(L,n_h2);
66 n           = 1;
67
68 if n_h2 > 0
69 for i = 1 : L-1
70     for j = i+1 : L
71         h2(i,[3*n-2  3*n-1  3*n])    = h2_block_1;
72         h2(j,[3*n-2  3*n-1  3*n])    = h2_block_2;
73
74         h2_name(n)     = {[num2str(i) '  ' num2str(j)]};
75         h2_name_ind(n) = 3*n-2 + n_h1;
76         n = n+1;
77     end
78 end
79     H2  = containers.Map(h2_name,h2_name_ind);
80 else
81     h2_name     = [];
82     h2_name_ind = [];
83     H2          = [];
84 end
85
86 clearvars i j n
87
88 % -- h3 --
89 h3          = zeros(L,n_h3);
90 n = 1;
91 if n_h3 > 0
92     for i = 1 : L-2
```

```
93          for j = i+1 : L-1
94              for k = j+1 : L
95                  h3([i j k],n) = 1;
96                  h3_name(n)      = {[num2str(i) '  ' num2str(j) '  '
                        num2str(k)]};
97                  h3_name_ind(n) = n + n_h1 + n_h2;
98                  n = n+1;
99
100             end
101         end
102     end
103     H3  = containers.Map(h3_name,h3_name_ind);
104 else
105     h3_name     = [];
106     h3_name_ind = [];
107     H3          = [];
108 end
109
110 HT  = containers.Map([h1_name h2_name h3_name],[h1_name_ind h2_name_ind
        h3_name_ind]);
111
112 h   = [h1 h2 h3];
113
114 SH  = fvar2struc(h,H1,H2,H3,HT,n_htot,a_coefname,b_coefname,a_coefnamevec
        ,b_coefnamevec,a_length,b_length,h1);
115
116 end
```

F-1-4 Write prescribed displacement fields static condensation

```
1  function [filename,B] = fSTEP_csv(n_eig, U_eig, qm,n_node_disp,
       n_field_disp,dof_dir,n_f,coord_disp)
2
3  % INPUT: n_eig, U_eig, qm,n_node,n_field_disp,dof_dir,n_f,coord_t
4
5  h_eig   = eye(n_eig);
6  Q_eig   = U_eig*qm*h_eig;       % combined displacement vector, all dir
7  X       = zeros(n_node_disp,n_field_disp*n_eig);
8
9  for j=1:n_eig
10     for i=1:n_field_disp
11         kol         = (j-1)*n_field_disp + i;
12         X(:,kol)    = Q_eig(dof_dir==n_f(i),j);
13     end
14 end
15
16 B           = [coord_disp X];
17 filename    = 'eigenvectors.csv';              % schrijf file in current
       folder
18
19 % toon max verplaatsing
20 disp_w_max  = max(X,[],1);
21
```

```
end
```

F-1-5 Import solution eigenvalue analysis from Comsol

```
function [MS] = fSTEP_sol(model,ind_eig,DIM)
% STORE SOLUTION IN STRUCTURE

n_eig           = length(ind_eig);
solution_info   = mphsolutioninfo(model,'soltag','sol1')';
                    % list of solution tags
sol_info        = mphsolinfo(model,'soltag','sol1');
n_dof           = sol_info.size;
n_sol           = sol_info.sizesolvals;
vec_all         = [1:n_sol];

U_eig           = mphgetu(model,'soltag','sol1','solnum',ind_eig);
                % solution matrix
U_all           = mphgetu(model,'soltag','sol1','solnum',vec_all);

freq            = abs( imag(sol_info.solvals)./(2*pi));
eigenval        = sol_info.solvals;

% filter on eigenvalues
%
%% FILTER EIGENVALUES, REMOVE NAN AND REAL eigenvalues (we added no
    damping)
ind_val     = find( ~isnan(eigenval) & real(eigenval) == 0 & abs(imag(
    eigenval)) > 1 ); % dit is de beste selectie

freq        = freq(ind_val);

%% EXTRACT EIGENVECTORS
U_eig       = mphgetu(model,'soltag','sol1','solnum',ind_val(ind_eig));
                % solution matrix
U_all       = mphgetu(model,'soltag','sol1','solnum',ind_val);
%}


[MS] = fvar2struc(sol_info,n_dof,n_sol,U_eig,U_all,freq,eigenval,n_eig);

end
```

F-1-6 Create parametric sweep in Comsol

```
% EXPORT h-MATRIX TO COMSOL

study_name  = ['std' num2str(study_nr)];
solut_name  = ['sol' num2str(study_nr)];

for i=1:size(h,1)
    h_name{1,i} = ['h' num2str(i)];
    p_name{1,i} = ['pAPI' num2str(i)];
```

```
9       h_val{1,i}  =  num2str(h(i,:));
10      p_unit{1,i} = '';
11  end
12
13  %% FILTER ON EXISTING PAR SWEEPS
14
15  % show par sweeps
16  model.sol(['sol' num2str(study_nr)]).feature('s1');
17
18  txt_out = char(model.sol(solut_name).feature('s1').toString());
19
20  ind_p   = strfind(txt_out,'pAPI1');
21  ex_p    = ~isempty(ind_p);
22
23  if ex_p == 1
24      model.sol(solut_name).feature('s1').feature.remove('pAPI1'); % dus
            bij feature moet je niet de naam opgeven!
25  else
26  end
27
28  %%
29
30  model.sol(['sol' num2str(study_nr)]).feature('s1').create('pAPI1', '
        Parametric');                   % standard name is pDef, maar de 2e par
        sweep heet p1.
31
32
33  model.study(study_name).feature('stat').set('pname', h_name);
34  model.study(study_name).feature('stat').set('plistarr', h_val);
35  model.study(study_name).feature('stat').set('punit', p_unit);
36
37  model.sol(solut_name).feature('v1').set('clistctrl', p_name);
38  model.sol(solut_name).feature('v1').set('cname', h_name);
39  model.sol(solut_name).feature('v1').set('clist', h_val);
40  model.sol(solut_name).feature('s1').set('probesel', 'none');
41  model.sol(solut_name).feature('s1').feature('pAPI1').set('pname', h_name)
        ;
42  model.sol(solut_name).feature('s1').feature('pAPI1').set('plistarr',
        h_val);
43  model.sol(solut_name).feature('s1').feature('pAPI1').set('punit', p_unit)
        ;
```

F-1-7 Linear algebra to determine nonlinear stiffness coefficients

```
1  function [A,B,a_coef,b_coef] = fSTEP_coef(SH,qv,F_nonlin)
2
3
4  % INPUT:    a_coefname, b_coefname, H1, H2, H3, qv, F_nonlin
5  % OUTPUT:   A,B,a_coef,b_coef
6
7  fstruc2var(SH)
8
```

```
9  a_set1      = a_coefname(a_coefname(:,1) == a_coefname(:,2),:);          %
       coeff names with all the same index
10 a_set1_ind  = find(a_coefname(:,1) == a_coefname(:,2));                  %
       index of combi's in a_coefname vector
11 a_set2      = a_coefname(a_coefname(:,1) ~= a_coefname(:,2),:);          %
       coeff names with unequal indices
12 a_set2_ind  = find(a_coefname(:,1) ~= a_coefname(:,2));                  %
       index of combi's
13
14 % 1e set: a_11
15 for i = 1:size(a_set1,1)
16     q1                          = qv(a_set1(i,1));
17     F1_ind                      = H1(num2str(a_set1(i)));
18     a_coef(a_set1_ind(i),1)     = { ( F_nonlin(:,F1_ind) + F_nonlin(:,
           F1_ind+1) ) ./ (2*q1^2) };
19
20 end
21
22 A       = containers.Map(a_coefnamevec(a_set1_ind),a_coef(a_set1_ind));
23
24 % 1e set b
25 b_set1      = b_coefname(~std(b_coefname')',:);          % coeff combi
       equal
26 b_set1_ind  = find(~std(b_coefname')');                  % index of combi'
       s
27
28 for i = 1:size(b_set1,1)
29     q1                          = qv(b_set1(i,1));
30     q2                          = qv(b_set1(i,2));
31     q3                          = qv(b_set1(i,3));
32     F1_ind                      = H1(num2str(b_set1(i)));
33     b_coef(b_set1_ind(i),1)     = { ( F_nonlin(:,F1_ind) - F_nonlin(:,
           F1_ind+1) ) ./ (2*q1^3) };
34 end
35
36 B       = containers.Map(b_coefnamevec(b_set1_ind),b_coef(b_set1_ind));
37
38 %% STOP IF ONLY a_11 COEFF ARE DESIRED
39 if exist('a_only')
40     if a_only == 1
41         return
42     else
43     end
44 else
45 end
46
47 %% CONTINUE NORMAL WAY
48
49
50 % 2e set a
51 for i = 1:size(a_set2,1)
52
53     q1                  = qv(a_set2(i,1));
```

```
54      q2                      = qv(a_set2(i,2));
55
56      F1_ind                  = H2(num2str(a_set2(i,:)));
                                      % index in de verpl matrix
57
58      f1                      = a_coefname(a_set2_ind(i),1);
59      f1_val                  = A(num2str([f1 f1]));
                                      % a_11 waarde
60
61      f2                      = a_coefname(a_set2_ind(i),2);
62      f2_val                  = A(num2str([f2 f2]));
                                      % zoek naa 22 als key voor a_22
            waarde
63
64      a_coef(a_set2_ind(i),:)     = {( F_nonlin(:,F1_ind) + F_nonlin(:,
            F1_ind+1) - 2*q1^2 .* ...
65                                      f1_val - 2*q2^2 .* f2_val) ./ (2*q1*
                                          q2)};
66  end
67
68  % write maps
69  keySet     = a_coefnamevec([a_set1_ind; a_set2_ind]);
70  valueSet   = a_coef(       [a_set1_ind; a_set2_ind]);
71  A       = containers.Map(keySet,valueSet);
72  clearvars keySet valueSet
73
74  % 2e set b: b_122
75  b_set2_bool = (b_coefname(:,2) == b_coefname(:,3)) & (b_coefname(:,1) ~=
        b_coefname(:,2) );
76  b_set2      = b_coefname(b_set2_bool,:);
77  b_set2_ind  = find(b_set2_bool);
78
79  % 3e set b: b_112
80  b_set3_bool = (b_coefname(:,1) == b_coefname(:,2)) & (b_coefname(:,1) ~=
        b_coefname(:,3) );
81  b_set3      = b_coefname(b_set3_bool,:);
82  b_set3_ind  = find(b_set3_bool);
83
84  % 2e set b: b_122
85  for i = 1:size(b_set2,1)
86      q1                      = qv(b_set2(i,1));
87      q2                      = qv(b_set2(i,2));
88
89      name                    = b_coefname(b_set2_ind(i),:);
                                      % coeff naam, indices
90      F1_ind                  = H2(num2str([name(1) name(2)]));
                                    % index in de verpl matrix
91      a_11                    = A( num2str([name(1) name(1)]));
                                    % a_11  waarde
92      a_22                    = A( num2str([name(2) name(2)]));
                                    % a_22  waarde
93      b_111                   = B( num2str([name(1) name(1) name(1)]));
                              % b_111 waarde
```

```
94
95      b_coef(b_set2_ind(i),:) = { ( F_nonlin(:,F1_ind) + F_nonlin(:,F1_ind
            + 2)  ...
96                              - 2*q1^2 .* a_11 - 2*q2^2 .* a_22 - 2*q1^3 .*
                                    b_111 ) ./ (2*q1*q2*q2) };
97  end
98
99  % 3e set b: b_112
100 for i = 1:size(b_set3,1)
101     q1                  = qv(b_set3(i,1));
102     q2                  = qv(b_set3(i,3));
103
104     name                = b_coefname(b_set3_ind(i),:);
                                  % coeff naam, indices
105     F1_ind              = H2(num2str([name(2) name(3)]));
                                % index in de verpl matrix
106     a_12                = A( num2str([name(1) name(3)]));
                                % a_11  waarde
107     b_222               = B( num2str([name(3) name(3) name(3)]));
                          % b_111 waarde
108
109     b_coef(b_set3_ind(i),:) = { ( F_nonlin(:,F1_ind) - F_nonlin(:,F1_ind
            + 2)  ...
110                             - 2*q1*q2 .* a_12 - 2*q2^3 .* b_222 ) ./ (2*
                                    q1*q1*q2) };
111 end
112
113
114 % write maps
115 keySet      = b_coefnamevec([b_set1_ind; b_set2_ind; b_set3_ind]);
116 valueSet    = b_coef(       [b_set1_ind; b_set2_ind; b_set3_ind]);
117 B           = containers.Map(keySet,valueSet);
118 clearvars keySet valueSet
119
120 % 4e set b: b_123
121 b_set4_bool =   (((b_coefname(:,1) ~= b_coefname(:,2)) & (b_coefname(:,1)
        ~= b_coefname(:,3))) & (b_coefname(:,2) ~= b_coefname(:,3)));
122 b_set4      = b_coefname(b_set4_bool,:);
123 b_set4_ind  = find(b_set4_bool);
124
125 format = '  %u  %u  %u';
126 % 4e set b: b_123
127 for i = 1:size(b_set4,1)
128     q1                  = qv(b_set4(i,1));
129     q2                  = qv(b_set4(i,2));
130     q3                  = qv(b_set4(i,3));
131
132     name                = b_coefname(b_set4_ind(i),:);
                                  % coeff naam, indices
133     F1_ind              = H3(num2str([name(1) name(2) name(3)],format));
                                % index in de verpl matrix
134     a_11                = A( num2str([name(1) name(1)]));
                                % a_11  waarde
```

```
135     a_22                = A( num2str([name(2) name(2)]));
                                % a_22  waarde
136     a_33                = A( num2str([name(3) name(3)]));
                                % a_22  waarde
137     a_12                = A( num2str([name(1) name(2)]));
                                % a_12  waarde
138     a_13                = A( num2str([name(1) name(3)]));
                                % a_12  waarde
139     a_23                = A( num2str([name(2) name(3)]));
                                % a_12  waarde
140     b_111               = B( num2str([name(1) name(1) name(1)],format));
                          % b_111 waarde
141     b_222               = B( num2str([name(2) name(2) name(2)],format));
                          % b_111 waarde
142     b_333               = B( num2str([name(3) name(3) name(3)],format));
                          % b_111 waarde
143     b_112               = B( num2str([name(1) name(1) name(2)],format));
                          % b_111 waarde
144     b_122               = B( num2str([name(1) name(2) name(2)],format));
                          % b_111 waarde
145     b_113               = B( num2str([name(1) name(1) name(3)],format));
                          % b_111 waarde
146     b_133               = B( num2str([name(1) name(3) name(3)],format));
                          % b_111 waarde
147     b_223               = B( num2str([name(2) name(2) name(3)],format));
                          % b_111 waarde
148     b_233               = B( num2str([name(2) name(3) name(3)],format));
                          % b_111 waarde
149
150
151     b_coef(b_set4_ind(i),:)  ={(F_nonlin(:,F1_ind) ...
152                               - a_11.*q1^2      - a_22.*q2^2        -
                                    a_33.*q3^2     - a_12.*q1*q2    ...
153                               - a_13.*q1*q3     - a_23.*q2*q3       -
                                    b_111.*q1^3                     ...
154                               - b_222.*q2^3     - b_333.*q3^3       -
                                    b_112.*q1*q1*q2                 ...
155                               - b_122.*q1*q2*q2 - b_113.*q1*q1*q3   -
                                    b_133.*q1*q3*q3                 ...
156                               - b_223.*q2*q2*q3 - b_233.*q2*q3*q3)  ./(q1
                                    *q2*q3) }   ;
157 end
158
159 keySet      = b_coefnamevec([b_set1_ind; b_set2_ind; b_set3_ind;
        b_set4_ind]);
160 valueSet    = b_coef(       [b_set1_ind; b_set2_ind; b_set3_ind;
        b_set4_ind]);
161 B           = containers.Map(keySet,valueSet);
162 clearvars keySet valueSet
163
164 end
```

F-2 Matlab-AUTO interface: Bifurcation analysis

F-2-1 Equations of motion written by Matlab

```
1         SUBROUTINE FUNC(NDIM,U,ICP,PAR,IJAC,F,DFDU,DFDP)
2         IMPLICIT DOUBLE PRECISION (A-H,O-Z)
3         DIMENSION U(NDIM),PAR(20)
4         DIMENSION F(NDIM),DFDU(NDIM,NDIM),DFDP(NDIM,20)
5
6         real*8 omega, TPI, pp, Nh,
7       * ft1,ft2 ,
8       * fp1,fp2 ,
9       * km1,km2 ,
10      * cm1,cm2
11        TPI=4*DATAN(1.0D0)
12        pp=PAR(3)/1000.d0
13        omega=PAR(2)
14        Nh = 0.0000d+00
15        ft1=-7.7934d-05 * PAR(1)
16        ft2=2.7989d-04 * PAR(1)
17        fp1 =0.0000d+00 * PAR(1)
18        fp2 =0.0000d+00 * PAR(1)
19        km1 = 1.0000d+00
20        km2 = 3.9263d+01
21        cm1 = 1.0000d-03
22        cm2 = 6.2660d-03
23        F(1)=U(2)
24        F(3)=U(4)
25        F(5)=U(5)+omega*U(6)-U(5)*((U(5))**2+(U(6))**2)
26        F(6)=-omega*U(5)+U(6)-U(6)*((U(5))**2+(U(6))**2)
27        F(2)=-km1*(1+fp1*U(5))*U(1)-cm1*U(2)+ft1*U(5)+eq1p(U)
28        F(4)=-km2*(1+fp2*U(5))*U(3)-cm2*U(4)+ft2*U(5)+eq2p(U)
29
30        IF(IJAC.EQ.0)RETURN
31        RETURN
32        END
33
34
35        function eq1p(U)
36        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
37        dimension U(6)
38        real*8 d1
39        d1 =
40      * -8.0862d-15*U(1)*U(1) -7.2646d-12*U(1)*U(3) -2.3874d-11*U(3)*U(3)
41      * -1.0000d+00*U(1)*U(1)*U(1) -6.3603d+00*U(1)*U(1)*U(3) -6.0204d+01*
          U(1)*U(3)*U(3)
42      * -6.3641d+01*U(3)*U(3)*U(3)
43
44        eq1p =
45      * +d1
46        return
47        end
```

```
48
49        function eq2p(U)
50        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
51        dimension U(6)
52        real*8 d1
53        d1 =
54       * -3.6323d-12*U(1)*U(1) -4.7747d-11*U(1)*U(3) --4.9852d-11*U(3)*U(3)
55       * -2.1201d+00*U(1)*U(1)*U(1) -6.0204d+01*U(1)*U(1)*U(3) -1.9092d+02*
           U(1)*U(3)*U(3)
56       * -3.8643d+02*U(3)*U(3)*U(3)
57
58        eq2p =
59       * +d1
60        return
61        end
62
63        SUBROUTINE STPNT(NDIM,U,PAR,T)
64        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
65        DIMENSION U(NDIM),PAR(20)
66
67        real*8 omega, ft1, TPI, pp
68        TPI=8*DATAN(1.0D0)
69        omega=0.6
70        pp=0.
71        ft1=0.
72        PAR(1)=ft1*10000.
73        PAR(2)=omega
74        PAR(3)=pp*1000.d0
75        PAR(11)=TPI/omega
76
77        do ii=1,4
78        U(ii)=0.
79        enddo
80        U(5)=DSIN(TPI*T)
81        U(6)=DCOS(TPI*T)
82        RETURN
83        END
84
85        SUBROUTINE BCND(NDIM,PAR,ICP,NBC,U0,U1,FB,IJAC,DBC)
86        RETURN
87        END
88        SUBROUTINE ICND(NDIM,PAR,ICP,NINT,U,UOLD,UDOT,UPOLD,FI,IJAC,DINT)
89        RETURN
90        END
91        SUBROUTINE FOPT(NDIM,U,ICP,PAR,IJAC,FS,DFDU,DFDP)
92        RETURN
93        END
94        SUBROUTINE PVLS(NDM,U,PAR)
95        RETURN
96        END
```

F-2-2 Useful AUTO commands

```
% COMPILATION FORTRAN PROJECT WORKSPACE
command    = [' echo off'...
              ' & cd C:\Users\Vincent\Documents\2. School\2017-2018\
                 Afstuderen\software\unzipped\Fortran + Duffing\FORTRAN\
                 FORTRAN\X86\DF\BIN'...
              ' & DFVARS.bat' ...
              ' & cd C:\Users\Vincent\Documents\2. School\2017-2018\
                 Afstuderen\software\unzipped\Fortran + Duffing\
                 Duffing_auto\Mostafa_auto' ...
              ' & NMAKE duffing.mak'];

% WRITE AUTO SETTINGS
fsettings(dfnr,NDIM,IRS,ICP,RL0,RL1,DS,DSMIN,DSMAX,NTST,autofolder)
% RUN AUTO
system(['echo off & cd ' autofolder '  & auto duffing 2 & newrun' ' & cd
    ' matlabfolder]);
```

F-2-3 Reading AUTO output

```
function [end_type, end_label, end_par, Mf7] = freadfort7(autofolder)
%{
input: autofolder
output: last point: type, label , par waarde
%}

fileID  = fopen([autofolder '\fort.7']);

fgetl(fileID);
tline       = fgetl(fileID);           % sla header & witregel over
i           = 0;

while feof(fileID) == 0
    tline       = fgetl(fileID);
    if ( any(tline(4) == '123456789') || any(tline(3) == '123456789') )
        i           = i + 1;
        tlinec      = split(tline);
        Mf7(i,:)    = str2double(tlinec(2:end))'; % eerst nam hij de
            branch niet mee
    end
end

fclose(fileID);

last_line       = Mf7(end,[2  3  4]);
end_type        = Mf7(end,3);
end_label       = Mf7(end,4);
end_par         = Mf7(end,5);
```

```
function  [Dat_f8] = freadfort8(n_labels,NTST,autofolder)

NCOL    = 4;
% NTST    = 40;
%% UITLEZEN FORT8
```

```
6
7  fileID  = fopen([autofolder '\fort.8']);
8
9  tline           = fgetl(fileID);
                                                    % sla header &
    witregel over
10 tlinec          = split(tline);
11 header_start    = tlinec(2:end);
12 ibr             = str2double(header_start{1}); %
13 ipnt            = str2double(header_start{2}); %
14 ptype           = str2double(header_start{3}); %
15 label           = str2double(header_start{4}); %
16 nfpr            = str2double(header_start{5}); %
17 isw             = str2double(header_start{6}); %
18 n_timepnt       = str2double(header_start{7}); % lusinfo
19 n_column        = str2double(header_start{8}); % lusinfo
20 nrowpr          = str2double(header_start{9}); % lusinfo
21 ntst            = str2double(header_start{10});
22 ncolloc         = str2double(header_start{11});
23 nparx           = str2double(header_start{12});
24 n_rest          = nrowpr - n_timepnt;        % sla nutteloze data over
25
26 frewind(fileID);                             % set position indicator back
    to begin of file
27 n_timepntmax    = NTST*NCOL + 1;             % max aantal tijdpunten per
    label
28 Uall            = zeros([n_timepntmax n_column n_labels]);
29 n_timepnt       = zeros(n_labels,1);         % aantal tijdstappen dat een
    label heeft
30 j   = 0;
31 while feof(fileID) == 0
32     j               = j + 1;
33     header(j,1:12)  = fscanf(fileID,'%d',[12,1])';
                                % 1e blok
34     n_timepnt(j)    = header(j,7);
35     Um              = fscanf(fileID,'%f',[n_column,n_timepnt(j)])';
36     Ummax(j,:)      = max(Um);
37     header(j,13:14) = fscanf(fileID,'%d',[2,1])';
                                 % 2e blok
38     par_dot(j,:)    = fscanf(fileID,'%f',[2,1])';
39     Um_dot          = fscanf(fileID,'%f',[n_column-1,n_timepnt(j)])';
40     Par(j,:)        = fscanf(fileID,'%f',[nparx,1])';
41     fgetl(fileID);
42
43     Uall(1:n_timepnt(j),:,j)     = Um;
44     clearvars  Um
45 end
46 fclose(fileID);
47
48 Dat_f8.Uall          = Uall;
49 Dat_f8.Par           = Par;
50 Dat_f8.n_timepnt     = n_timepnt;
51 Dat_f8.header        = header;   % 1e kol branch, 4e kol label
```

```
52
53 end
```

F-3 Matlab-Fortran

F-3-1 Create random initial conditions for specified energy levels

```
1  function [U0,v0,u0,MRI] = fode_initial(V0_amp,Vstart,T_noise,Eta,phase,
       CO1,MP,kB,mass_tot,MD,scale_T,scale_U)
2  % CREATE SET OF INITIAL CONDITIONS
3
4  %{
5  INPUT
6      OPTION 1: T_noise & Eta
7          Noise temperatuur & Eta
8
9      OPTION 2: T_noise & V0_amp
10
11     OPTION 3: V0_amp & Eta
12
13     OPTION 4: Vstart & Eta
14         een vaste begin-energie,
15         moment gedurende de val gespecificeerd door Eta
16 %}
17
18
19 fstruc2var(MP);
20 fstruc2var(CO1);
21
22 ndim        = n_eig*2;
                                                    % tbv ringdown
       ode solver
23 i_u         = 1:2:ndim;
24 i_v         = i_u + 1;
25 i_high      = i_v(2:end);
26
27 % INPUT
28 if ~isnan(Vstart) && ~isnan(Eta)          % OPTIE 4
29     V0i = Vstart *thic* omega_fm;
30     E0i = 1/2 * CO1.Mm_value(1) * V0i^2;% determine energy at start
31     E0  = Eta*E0i;                        % correct for already lost
           fraction
32     V0  = sqrt(2*E0/CO1.Mm_value(1));  % determine initial velocity from
           energy
33
34 elseif isnan(Vstart) && ~isnan(V0_amp)  % OPTIE 2,3,4
35     V0  = V0_amp *thic* omega_fm;
36     E0  = 1/2 * CO1.Mm_value(1) * V0^2;
37
38 end
39
40 N_noise = n_eig - 1;
```

```
41
42 if ~isnan(T_noise) && ~isnan(V0_amp)    % OPTIE 2
43
44     E_noise_per = kB*T_noise;
45     E_noise     = N_noise*E_noise_per;
46     E_tot       = E0 + E_noise;
47     Eta         = E0/E_tot;
48
49 elseif ~isnan(T_noise) && ~isnan(Eta)   % OPTIE 1
50
51     E_noise_per = kB*T_noise;
52     E_noise     = N_noise*E_noise_per;
53     E0          = E_noise_per;
54     V0          = sqrt(2*E0 / CO1.Mm_value(1));
55     V0_amp      = V0 / (thic* omega_fm);
56
57 elseif ~isnan(V0) && ~isnan(Eta) && isnan(T_noise)    % OPTIE 3
58
59     E_noise     = (1/Eta -1)*E0;
60     E_noise_per = E_noise/N_noise;
61     T_noise     = E_noise_per/kB;
62 else
63     disp('~~ no proper inital condiations. Choose T or Eta ~~')
64     return
65 end
66
67 % BEPAAL INITIAL SPEED NOISE MODES
68 m_noise     = CO1.Mm_value(2:end);
69 Vh          = sqrt(2*E_noise_per ./ m_noise);
70
71 %% RANDOM PHASE INITIAL CONDITIONS
72
73 % 0. vector eigenfrequencies
74 wn      = sqrt( CO1.Km_value ./ CO1.Mm_value );
75
76 % 2a. get nonlinear stiffness
77 [a11,b111] = fsolostiff(CO1);
78
79 % 2b. determine amplitude more precise
80 a               = 1/4.*b111 .* sin(phase).^4;
81 b               = 1/2.*CO1.Km_value .*  sin(phase).^2 + 1/2.*CO1.Mm_value
       .* wn.^2 .*  cos(phase).^2;
82 c               = -E_noise_per;
83
84 for i = 1:n_eig
85     rr          = roots([a(i) b(i) c]);
86     amp(i,1)     = real(sqrt(max(rr)));
87 end
88
89 % 3. create displacement & velocity
90 u       = amp.*    sin(phase);
91 v       = amp.*wn.*cos(phase);
92
```

```
93  % 4. write total vector
94  U01          = zeros(n_eig*2,1);
95  U01(i_u)     = u;
96  U01(i_v)     = v;
97  U01(i_v(1)) = V0;
98  U01(i_u(1)) = 0;
99
100 % 4. check total energy
101 [Es,~,~,~,~,~]  = fget_energy_mode(0,U01');
102
103 Emode           = Es.mode.t.T;
104 Enoise_i        = sum(Emode(2:end));
105 err             = (Enoise_i - E_noise) / E_noise;
106
107 %% BEPAAL DIM LESS ENERGY
108 E_tot       = E0 + E_noise;
109 MP.E_pre    = 2*eps_pre^2 *C_L^2 *mass_tot;
110 E_tilde_tot = E_tot / MP.E_pre;
111 E_tilde_eq  = E_tilde_tot / n_eig;
112
113 % % BEGIN MET MODE 1 SNELHEID
114 % U01          = zeros(n_eig*2,1);
115 % U01(i_v(1))  = V0;
116 % U01(i_high)  = Vh;
117
118 U0m          = U01 ./ MD.alpha2';       % --> transform to mass-
    normalised eigenvectors
119 U0           = U0m * scale_T / scale_U;  % scale to original domain
120
121 u0              = U0(i_u)'; % for text file
122 v0              = U0(i_v)';
123
124 zeta_ring       = 1/2 * C01.Cm_value(1) / sqrt(C01.Km_value(1) .* C01.
    Mm_value(1));
125
126 % write table
127 tab_name        = {'V0_amp';'E_tilde_tot'; 'E_tilde_eq';'T_noise';'Eta';'
    n_eig';'zeta' };
128 tab_val         = { V0_amp ; E_tilde_tot ;  E_tilde_eq ; T_noise ; Eta ;
    n_eig ; zeta_ring };
129 varNames        = {'Parameter' 'Value'};
130 Tab             = table(tab_name,tab_val,'VariableNames',varNames);
131
132 [MRI]           = fvar2struc(V0_amp,U0,T_noise,E_tilde_tot,Eta,E_tilde_eq
    ,E0,E_noise,zeta_ring,Tab);
133
134
135 end
```

F-3-2 Run ODE solvers in parallel

```
%% ODE PARALLEL

% OVERHEAD SETTINGS
varvec  = [1]';
n_var   = length(varvec);
n_par1  = 80; % sim per set

v_p1    = ones(n_par1,1);
v_tot   = v_p1*varvec';
v_vec   = reshape(v_tot,[],1);

% PARRALLEL SETTINGS
V0_amp  = 2;
Vstart  = NaN;
T_noise = 1 %5e5;
Eta     = NaN ;%1/n_eig
n_par   = n_par1*n_var;
T_vec   = v_vec;
xas_vec = varvec;

% show initial conditions
solo    = 1; % if solo == 1: no coupling energy
fwrite_energy_mode(CO1,solo);
phase   = rand([n_eig,n_par1]); %  create random phase for each mode 2*pi
    *

[U0,v0,u0,MRI]  = fode_initial(V0_amp,Vstart,T_noise,Eta,phase(:,1),CO1,
    MP,kB,mass_tot,MD,scale_T,scale_U);

warning('off', 'MATLAB:MKDIR:DirectoryExists');
% 1. create set of initial conditions: select changed parameter
for i = 1:n_par
    i_ph     = 1+ mod(i,n_par1);
    % create initial conditions
    T_noise              = T_vec(i);
    [U0(:,i),v0,u0,MRIp{i}]  = fode_initial(V0_amp,Vstart,T_noise,Eta,
        phase(:,i_ph),CO1,MP,kB,mass_tot,MD,scale_T,scale_U);
    % create folder
    status = mkdir([ode_folder '\set_' num2str(i)]);
    path_set            = [ode_folder '\set_' num2str(i)];
    path_p              = [path_set '\p.dat'];

    dlmwrite([path_set '\p_in.dat'],[v0;u0],'delimiter', ' ','newline', '
        pc');
    copyfile([path_fort_ode '\time_panel_Farbod.exe'],path_set);      %
        3. copy .exe to folders

end

n_par_max   = 40;
n_cpu       = 5;
n_check     = 0;
n_cmd       = 0;
```

```
49 n_go         =  n_par_max - n_cmd
50
51 for i = 1:n_par
52
53     path_set    = [ode_folder '\set_' num2str(i)];
54     system(['echo off & cd ' path_set '  & start /min time_panel_Farbod &
           cd ' matlabfolder ]); %run ode solver fortran
55     disp(['running ode ' num2str(i) ' out of ' num2str(n_par)])
56     n_go    = n_go - 1;
57
58     if n_go <= 0
59         pause(0.1)
60         [~, tasklist]   = system('tasklist');
61         n_cmd           = length(regexp(tasklist,'time_panel_Farbod.exe')
               );
62         n_go            =  n_par_max - n_cmd
63         if n_cmd > n_cpu
64             n_par_max = max(n_cpu,n_par_max - 2)
65         end
66     else
67     end
68
69     while n_cmd >= n_par_max
70         n_check         = n_check + 1;
71         [~, tasklist]   = system('tasklist');
72         n_cmd           = length(regexp(tasklist,'time_panel_Farbod.exe')
               );
73         n_go            =  n_par_max - n_cmd
74         pause(0.1)
75         end
76 end
77
78 while n_cmd > 0
79     [~, tasklist]   = system('tasklist');
80     n_cmd           = length(regexp(tasklist,'time_panel_Farbod.exe'));
81     pause(2)
82 end
```

```
1  %% FUNCTION FOR LOSS FACTOR ESTIMATE
2  %
3  f01     = sqrt(CO1.Km_value(1)/CO1.Mm_value(1))      / (2*pi);
4  fe1     = 10*f01;
5  Ts      = T/(2*pi);
6  f0      = 1/2*pi;
7
8  % 1. calc energy
9  solo    = 1; % if solo == 1: no coupling energy
10 fwrite_energy_mode(CO,solo);
11
12 % per simulation
13 for i = 1:n_par
14     [Es,~,~,~,~,~]  = fget_energy_mode(T,Yp{i});
15
```

```
    Emode{i,1}   = Es.mode.t.T;
    Enoise(:,i) = sum(Es.mode.t.T(:,2:end),2);
    E1(:,i)      = Es.mode.t.T(:,1);
    Etot(:,i)    = sum(Es.mode.t.T,2);

    % 2. calc fft
    [omega(:,i)]= fEN_fft(T,Yp{i}(:,1),f0,10*f0);

    % 2. calc loss factor
    [NEta(:,i)]  = fEN_loss_facor(Emode{i},T,omega(:,i),n_pnt);
end

%% exctract Eta at start
per_set         = 20;

for i = 1:n_var
    i_set(:,i)       = (i-1)*n_par1 + [1:n_par1];
    NEta_set(:,:,i) = NEta(:,i_set(:,i));
    [MRS(i)]         = fENeta_stat(NEta_set(:,:,i),per_set,n_pnt);

    Eta_var(i) = MRS(i).Eta_mean;
end
```

F-4 Energy

F-4-1 Modal energy

```
function [ET] = fenergy_mode(y)
km1 = 467.29;
km2 = 18351;
mm1 = 0.11775;
mm2 = 0.11778;
cm1 = 0;
cm2 = 0;
a1_1_1 = 8.4661e-12;
a2_1_1 = 1.9017e-09;
a1_1_2 = 3.8034e-09;
a2_1_2 = 2.5e-08;
a1_2_2 = 1.25e-08;
a2_2_2 = -5.221e-08;
b1_1_1_1 = 2345.8;
b2_1_1_1 = 2487;
b1_1_1_2 = 7460.9;
b2_1_1_2 = 70628;
b1_1_2_2 = 70628;
b2_1_2_2 = 2.24e+05;
b1_2_2_2 = 74668;
b2_2_2_2 = 9.0685e+05;
ET(1)       = +1/3 *a2_2_2* y(3)* y(3)* y(3) +1/4 *b2_2_2_2* y(3)* y(3)* y
    (3)* y(3) +1/2*km2*y(3)^2 +1/2*mm2*y(4)^2;
ET(2)       = +1/3 *a1_1_1* y(1)* y(1)* y(1) +1/4 *b1_1_1_1* y(1)* y(1)* y
    (1)* y(1) +1/2*km1*y(1)^2 +1/2*mm1*y(2)^2;
```

```
24 ET(3)       = +1/1 *a2_1_1* y(3)* y(1)* y(1) +1/2 *a1_1_2* y(1)* y(1)* y
       (3) +1/2 *a2_1_2* y(3)* y(1)* y(3) +1/1 *a1_2_2* y(1)* y(3)* y(3) +1/1
        *b2_1_1_1* y(3)* y(1)* y(1)* y(1) +1/3 *b1_1_1_2* y(1)* y(1)* y(1)* y
       (3) +1/2 *b2_1_1_2* y(3)* y(1)* y(1)* y(3) +1/2 *b1_1_2_2* y(1)* y(1)*
        y(3)* y(3) +1/3 *b2_1_2_2* y(3)* y(1)* y(3)* y(3) +1/1 *b1_2_2_2* y
       (1)* y(3)* y(3)* y(3);
25 end
```

```
1 function [ETname] = fenergy_mode_name()
2 ETname{1}  = [2];
3 ETname{2}  = [1];
4 ETname{3}  = [1   2];
5 end
```

F-4-2 Calculate loss factor

```
1  function [Eta] = fEN_loss_facor(Emode,T,omega,n_pnt)
2  % CALCULATE ENERGY & LOSS FACTOR
3
4  %{
5  INPUT:
6      Emode:  Energy per mode
7      omega:  Freq of vibration fund mode
8
9  OUTPUT:
10     Eta:    Loss factor
11 %}
12
13 dtT     = T(2) - T(1);
14 n_tot   = length(T);
15 wind    = n_pnt*5;
16
17 E1_ruw  = Emode(:,1);
18 E1      = smoothdata(E1_ruw,'movmean',wind);
19
20 P1_ruw  = -1.*diff(E1) ./ dtT;
21 P1      = smoothdata([P1_ruw ; NaN],'movmean',wind);
22
23 w_smooth= smoothdata(omega,'movmean',wind);
24
25 Eta     = P1 ./ (E1 .* w_smooth);
26
27 end
```

Bibliography

[1] L. N. Virgin, *Vibration of Axially Loaded Structures*. Cambridge University Press, 2007.

[2] A. Nayfeh and D. Mook, *Nonlinear Oscillations*. Wiley Classics Library, Wiley, 2008.

[3] A. Abolfathi, M. Brennan, and T. Waters, "Large deflection of a simply supported beam." August 2010.

[4] D. Midtvedt, A. Croy, A. Isacsson, Z. Qi, and H. S. Park, "Fermi-pasta-ulam physics with nanomechanical graphene resonators: Intrinsic relaxation and thermalization from flexural mode coupling," *Phys. Rev. Lett.*, vol. 112, p. 145503, Apr 2014.

[5] C. TOUZÉ, O. THOMAS, and A. CHAIGNE, "Asymmetric non-linear forced vibrations of free-edge circular plates. part 1: Theory," *Journal of Sound and Vibration*, vol. 258, no. 4, pp. 649 – 676, 2002.

[6] K. Markestein, "Finite element based analysis and validation for nonlinear structural dynamics." August 2018.

[7] A. A. Muravyov and S. A. Rizzi, "Determination of nonlinear stiffness with application to random vibration of geometrically nonlinear structures," *Computers and Structures*, vol. 81, no. 15, pp. 1513 – 1523, 2003.

[8] M. P. Mignolet, A. Przekop, S. A. Rizzi, and S. M. Spottswood, "A review of indirect/non-intrusive reduced order modeling of nonlinear geometric structures," *Journal of Sound and Vibration*, vol. 332, no. 10, pp. 2437 – 2460, 2013.

[9] L. Wu and P. Tiso, "Nonlinear model order reduction for flexible multibody dynamics: a modal derivatives approach," *Multibody System Dynamics*, vol. 36, 09 2015.

[10] K. Kim, V. Khanna, X. Wang, and M. Mignolet, *Nonlinear Reduced Order Modeling of Flat Cantilevered Structures*.

[11] J. Hollkamp, R. Gordon, and S. Spottswood, "Nonlinear modal models for sonic fatigue response prediction: A comparison of methods," *Journal of Sound and Vibration*, vol. 284, pp. 1145–1163, 06 2005.

[12] M. MCEWAN, J. WRIGHT, J. COOPER, and A. LEUNG, "A combined modal/finite element analysis technique for the dynamic response of a non-linear beam to harmonic excitation," *Journal of Sound and Vibration*, vol. 243, no. 4, pp. 601 – 624, 2001.

[13] D. Davidovikj, F. Alijani, S. J. Cartamil Bueno, H. Zant, M. Amabili, and P. Steeneken, "Nonlinear dynamic characterization of two-dimensional materials," *Nature Communications*, vol. 8, 12 2017.

[14] W. Greiner, *Classical Mechanics*. Springer, 2010.

[15] S. Rao, *Mechanical vibrations*. Pearson, 2011.

[16] F. Verhulst, *Nonlinear Differential Equations and Dynamical Systems*. Springer, 2000.

[17] V. Ramakrishnan and B. F. Feeny, "Resonances of a Forced Mathieu Equation With Reference to Wind Turbine Blades," *Journal of Vibration and Acoustics*, vol. 134, 10 2012. 064501.

[18] R. Dolleman, S. Houri, A. Chandrashekar, F. Alijani, H. Zant, and P. Steeneken, "Opto-thermally excited multimode parametric resonance in graphene membranes," *Scientific Reports*, vol. 8, 12 2018.

[19] F. Yang, F. Rochau, J. S. Huber, A. Brieussel, G. Rastelli, E. M. Weig, and E. Scheer, "Spatial modulation of nonlinear flexural vibrations of membrane resonators," *Phys. Rev. Lett.*, vol. 122, p. 154301, Apr 2019.

[20] J. Güttinger, A. Noury, P. Weber, A. Eriksson, C. Lagoin, J. Moser, C. Eichler, A. Wallraff, A. Isacsson, and A. Bachtold, "Energy-dependent path of dissipation in nanomechanical resonators," *Nature nanotechnology*, vol. 12, 05 2017.

[21] E. Fermi, P. Pasta, S. Ulam, and M. Tsingou, "Studies of the nonlinear problems," 5 1955.

[22] A. CARCATERRA, "An entropy formulation for the analysis of energy flow between mechanical resonators," *Mechanical Systems and Signal Processing*, vol. 16, no. 5, pp. 905 – 920, 2002.

[23] R. Barton, B. Ilic, A. Zande, W. Whitney, P. Mceuen, J. Parpia, and H. Craighead, "High, size-dependent quality factor in an array of graphene mechanical resonators," *Nano letters*, vol. 11, pp. 1232–6, 02 2011.

[24] H. B. Callen and T. A. Welton, "Irreversibility and generalized noise," *Phys. Rev.*, vol. 83, pp. 34–40, Jul 1951.

[25] J. Gutierrez Vega, R. Rodríguez-Dagnino, A. Meneses, and S. Chávez-Cerda, "Mathieu functions, a visual approach," *American Journal of Physics*, vol. 71, pp. 233–242, 03 2003.

[26] M. Bereyhi, A. Beccari, S. Fedorov, R. Schilling, D. Wilson, N. J. Engelsen, and T. Kippenberg, "Clamp-tapering increases the quality factor of stressed nanobeams," 09 2018.

[27] D. Jordan and P. Smith, *Nonlinear Ordinary Differential Equations: An Introduction for Scientists and Engineers*. Nonlinear Ordinary Differential Equations: An Introduction for Scientists and Engineers, OUP Oxford, 2007.

[28] C. de Silva, *Vibration Damping, Control, and Design*. CRC Press, 2007.

Glossary

List of Acronyms

FEM	Finite Element Modelling
ROM	Reduced Order Model
OOP	out-of-plane
STEP	Stiffness Evaluation Procedure
ODE	Ordinary differential equation
API	Application programming interface
dof	degree of freedom
IMSL	International Mathematics and Statistics Library
BDF	Backward differentiation formula
ASD	Amplitude Spectral Density
FPU	Fermi-Pasta-Ulam