# Department of Precision and Microsystems Engineering

Topology optimization involving constrained eigenfrequencies

Menno Huigsloot

Report no   : 2018.018
Coach   : Sasan Keyvani Janbahan, Arnoud Delissen
Professor   : Matthijs Langelaar
Specialisation   : Structural Optimization and mechanics
Type of report   : Master thesis
Date   : July 19 2018

TU Delft   Delft
University of
Technology
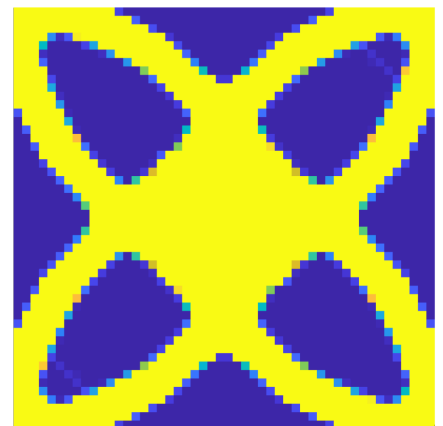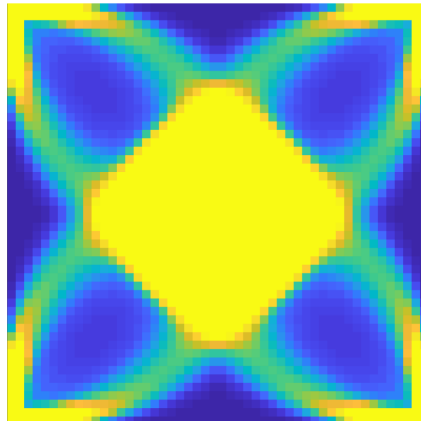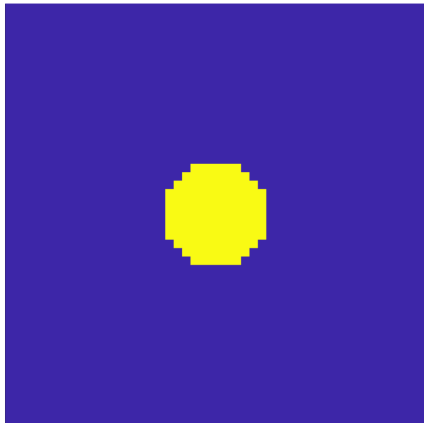
Topology optimization involving constrained eigenfrequencies

Master Thesis

By

Menno Huigsloot

in partial fulfilment of the requirements for the degree of Master of science
at Delft University of Technology,
to be defended publicly on Thursday, July 19 2018 at 14:00.

Student number: 4110587
Project duration: September 1, 2016   July 19, 2018
Thesis committee: Dr. Ir. M. Langelaar   TU Delft
                    Dr. Ir. Martin van Gijzen   TU Delft
                    Ir. A. Keyvani   TU Delft / TNO
                    Ir. A.A.T.M. Delissen   TU Delft

TUDelft
Delft
University of
Technology

Abstract

High performance machines such as those used in the semiconductor industry, robotics or racing engines have lots of fast moving parts. The dynamic properties of these moving parts are crucial to the performance of the machine. Therefore these moving parts have to be carefully designed which is often a very time consuming iterative process. In this thesis a general method to optimize the dynamic properties of a structure utilizing topology optimization is investigated. More specifically, the method will be focused on the optimization of eigenfrequencies whilst achieving specific ratios between eigenfrequencies, as dynamic performance requirements are often linked to such criteria. We refer to this class of topology optimization problems as problems involving constrained eigenfrequencies. A particular case of interest is the desired multiplicity of two or more eigenfrequencies, that is a ratio of 1.

Several crucial aspects of the topology optimization of eigenfrequencies are investigated, these are the material interpolation methods, mode tracking techniques, multiplicity problems and obtaining a discrete design. By comparing different material interpolation methods, a clear view on the effects of different methods is obtained, leading to solid arguments for selecting a linear material interpolation method for topology optimization of eigenfrequencies. A simple yet effective method of tracking the eigenmodes during the optimization process combined with a solution for the multiplicity problems is presented and verified to show similar results as a more complex analytical approach. A new method of obtaining a discrete design without applying penalization or modification of the eigenvalue problem has been developed using a modified objective function. This method shows promising results and is a good candidate for replacing the material interpolation penalization method. By combining these results, a general and capable framework for the topology optimization of constrained eigenfrequencies is obtained. Using the presented framework, a practical application of the method is given by the design of a cantilever used in an atomic force microscope. Feasible and well-performing designs have been generated, both from a functional and manufacturing point of view.

Preface

Before you lies the thesis "Topology optimization of constrained eigenfrequencies", the basis of which is my interest in creating the best possible designs. It has been written to fulfill the graduation requirements of the master Mechanical Engineering at the Delft university of technology. I was engaged in researching and writing this thesis from September 2016 to July 2018.

The thesis subject was selected by sharing my interests in dynamics and optimization with my supervising professor, Matthijs Langelaar. He provided me with some subjects based on these interests and the choice was made almost instantly. The research was a bit unclear initially as I was doing my literature review but started to take form when I started experimenting with my own topology optimization code. But most of all I really enjoyed the experimenting, developing the code and increasing the complexity of the problems I was facing. Early on in the project Sasan Keyvani joined as my daily supervisor, bringing some exciting ideas for the practical application to the table which resulted in the case studies shown in this thesis.

I would like to thank Matthijs and Sasan, for their guidance and interesting meetings during this process. I also wish to thank Arnoud Delissen for joining the later stages of the project and supporting me with writing this thesis.

During my graduation project I have also had the opportunity to represent Taylor, the study association of the Precision and Microsystems engineering department. Together with Thijs Blad, Miranda Looman, Niek van Hoek and Maarten de Jager we represented the department all over the world for a year. All the trips and events we participated in and organized throughout the year have been very motivating for me. For that, I would like to thank all of you.

I hope you enjoy your reading.

Menno Huigsloot
Delft, July 2018

Contents

List of Figures

List of Tables

Nomenclature

Symbols

$\bar{E}_{ijkl}$	Elasticity tensor
p	Penalization factor
ρ	Design density variable
ρ_e	Element design density variable
ρ_{min}	Minimum design density value
$\tilde{\rho_e}$	Filtered element density
ω	Frequency ($\mathrm{rad\,s^{-1}}$)
P_k	Stiffness penalization factor
P_m	Mass penalization factor
K	Stiffness matrix of full model
M	Mass matrix of full model
k_e	Element stiffness matrix obtained from ANSYS
m_e	Element mass matrix obtained from ANSYS
n	Number of elements in a mesh
ϕ_i	ith mode shape
ϕ_r	Reference mode shape
ϕ^k	K^{th} iteration mode shapes
δ_{ij}	Kronecker delta
E_p	Potential energy
E_k	Kinetic energy
γ	Modal stiffness
μ	Modal mass
f_k	Stiffness interpolation function
f_m	Mass interolation function
ϵ	Minimum value used in Pedersen's functions
r	Ratio between linear and power law in polynomial interpolation
λ_j	J$^{\text{th}}$ eigenvalue
$\Delta\lambda_{ij}$	Gap between ith and jth eigenvalue
r_{ij}	Ratio of the ith and jth eigenvalue
F	Load vector
u	Displacement vector
C	Compliance
C_{max}	Maximum compliance
k_i	Desired stiffness of degree of freedom i
N_e	Neighborhood of element e
R	Neighborhood filter radius
x_i	Geometrical location of the center of element i
$w(x_i)$	Location based weighting function
v_e	Element volume
β	Heaviside step function parameter
c_i	Component of ϕ_i in the linear combination
ν	Poisson's ratio
E	Young's modulus

V Design volume
V_{max} Relative maximum design volume
V_{min} Relative minimum design volume
η Relaxation parameter
α Discreteness value continuation parameter

Acronyms

ESO   Evolutionary structural optimization
BESO   Bi-directional evolutionary structural optimization
LSM   Level set method
SIMP   Solid isotropic material with penalization
MAC   Modal assurance criterion
FEM   Finite element method
MMA   method of moving asymptotes
DOF   Degrees of freedom
RAMP   Ramp interpolation method
MND   Measure of non-discreteness
AFM   Atomic force microscope

<div style="text-align: right;">

1

Introduction

</div>

High performance machines such as those used in the semiconductor industry, robotics or racing engines have lots of fast moving parts. The dynamic properties of these moving parts are crucial to the performance of the machine. Therefore these moving parts have to be carefully designed which is often a very time consuming iterative process. In this thesis a general method to optimize the dynamic properties of a structure utilizing topology optimization is investigated. More specifically, the method will be focused on the optimization of eigenfrequencies whilst achieving specific ratios between eigenfrequencies, as dynamic performance requirements are often linked to such criteria. A particular case of interest is the desired multiplicity of two or more eigenfrequencies, that is a ratio of 1. A practical application of the method is given by the design of a cantilever used in an atomic force microscope (AFM) first described by Binnig et al. [Binnig et al., 1986].

To clarify the design challenge of an AFM cantilever, first AFM itself will be briefly introduced. The AFM uses the tip of a cantilever to scan the surface of a substrate in order to generate a 3D image of the surface on an atomic scale. A simple overview of the AFM is shown in Figure 1.1 along with its most common operating modes. In contact mode the tip is dragged along the surface of a substrate and the position of the tip is constantly measured with a laser to generate a 3D image. By exciting the cantilever in this mode the stiffness of the substrate can also be measured. In the tapping or non-contact mode the first bending eigenmode of the cantilever is excited. In both of these modes the dynamic properties of the cantilever have a big impact on the measurement accuracy and resolution. By optimizing the dynamic properties of the cantilever the measurements can be improved.
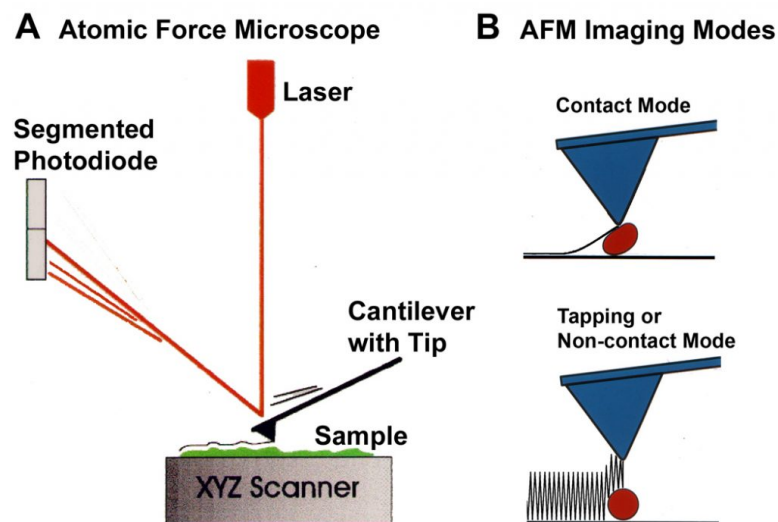
Figure 1.1: Simple overview of the atomic force microscope and its operating modes. Source: [CCEM, 2018]

Both of these operating modes require a different design of the cantilever. The method developed in this thesis will be used to optimize a cantilever for both configurations. The cantilever for the contact mode can be improved by maximizing the eigenfrequency of the first bending mode and can be improved even further by creating a multiplicity of the first bending and torsion mode. This will result in a higher sensitivity of the tip, which will in turn result in improved measurements.

The cantilever used in the tapping or non-contact mode can also be improved by maximizing the eigenfrequency of the first bending mode. It can be improved even further by making use of the so-called superharmonics, these are eigenfrequencies with an integer ratio with respect to the first eigenfrequency. Exciting this cantilever at the frequency of the first eigenmode will result in excitation of all the superharmonic modes. This allows the tip of the cantilever to follow the surface even more precisely and thus improve measurements as shown by Bongwon et al. [Bongwon et al., 2016].

General methods to optimize the eigenfrequencies of a structure have already been developed to some extent. One of the first methods presented by Ma et al. [Ma et al., 1994], is able to optimize the mean eigenvalue of several eigenmodes. The optimization of the fundamental eigenfrequency of a structure is shown in a book by Bendsoe and Sigmund [BendsØe and Sigmund, 2003], along with references to many different methods for the topology optimization of eigenvalue problems. These methods and more recently published methods have the optimization of eigenfrequencies in common. The main problem of these methods is the non-smoothness of eigenvalues when multiplicity occurs. A method to optimize the fundamental frequency of a structure with multiplicity was presented by Du and Olhoff [Du and Olhoff, 2005], however this method requires the use of a sub-optimization routine at every iteration, which leads to a high computational cost.

The optimization of the AFM cantilever has also been done before. An early paper presented by Pedersen [Pedersen, 2000], shows the optimization of the fundamental eigenfrequency of the cantilever. The frequency of all other modes is constrained to be at least 6 times higher while also constraining the static stiffness of the tip. A different method presented by Xia et al. [Xia et al., 2014] is able to maximize the surface area of a cantilever while applying a constraint on the ratio of the first and second bending mode. All of the separate parts required to optimize the design of a cantilever are presented in literature, however they have not been combined yet. Besides that, several challenges remain as discussed below, which make that further research into topology optimization of eigenfrequencies is needed.

Even though the AFM case studies are quite specific, the scope of this research thesis is the development of a general method to optimize the eigenfrequencies of a structure with constraints on eigenfrequencies and investigate ways to address known challenges of topology optimization for eigenvalue problems. These challenges are:

- Selecting a suitable material interpolation method that results in stable convergence.
- Dealing with the non-smoothness of the eigenvalues and mode tracking when multiplicity occurs.
- Handling mode switching during the optimization process due to the instability that it can cause.
- Converging towards a manufacturable discrete design.

With regard to these challenges, the present state of the art can be summarized as follows.

The material interpolation methods that are commonly used are derived from the optimization of static problems, which is why they cause issues when used to optimize the dynamic properties of a structure. Localized eigenmodes as shown by Pedersen [Pedersen, 2000], is one of these issues and is seen as a problem when optimizing eigenfrequencies. This is usually avoided by certain heuristic measures, but there is a lack of understanding of these methods. Besides the problems that arise, there is also a lack of information concerning the choice of material interpolation method when it comes to penalization of intermediate densities. Many different methods are available in literature but there seems to be no clear view on which method works best and why.

Multiplicity of eigenmodes is a subject that is often avoided in literature due to its difficulties. The biggest difficulty lies with the computation of the sensitivities as shown by Seyranian [Seyranian et al., 1994]. These problems are often avoided by applying a constraint on the order of eigenmodes as shown by Tsai and Cheng [Tsai and Cheng, 2013], preventing multiplicity. But since a constraint is applied, this is a restrictive method that might result in a sub-optimal design. There is a need for an effective yet practical method to deal with multiplicity during the optimization. This can be in the case when two modes switch order or when multiplicity is actually desired as in the case of the AFM.

Different methods are available to track the order of the modes during the optimization process, for example the use of the MAC-value as shown by Kim and Kim [Kim and Kim, 2000]. The MAC-value represents the projection of the new mode shape on the reference mode shape, and is used to compare the modeshape of the previous and current iteration. When multiplicity occurs during mode switching this becomes problematic, since the eigenmodes are no longer uniquely defined and any combination of the eigenmodes can be the obtained from the eigenvalue computation. As this method relies on the modeshape, it can no longer track the eigenmodes correctly once multiplicity occurs.

Obtaining a manufacturable discrete design from topology optimization is crucial for the practical implementation of the method. This is directly linked to the choice of material interpolation method and filters or constraints that are applied. Approaches to enforce manufacturability have been developed for topology optimization, but these have not been studied systematically in combination with eigenfrequency optimization.

The main objective of this research thesis is the development of a robust framework for the topology optimization of constrained eigenfrequencies. This framework has several crucial components, these will be investigated and their results will be implemented into the framework. The goals related to these components are:

- Analysing and finding solutions for common problems in topology optimization of eigenvalues related to the material interpolation method.
- Finding a simple solution to deal with mode tracking and multiplicity.
- Finding a suitable method to obtain a manufacturable discrete design.

The developed framework is used to optimize the case studies of the AFM cantilever. The goals related to these case studies are:

- Optimizing a structure with desired multiplicity of selected modes.
- Optimizing a structure with desired integer ratios of eigenfrequencies of selected modes.

Chapter 2 of this thesis will introduce the topology optimization method used throughout this thesis. All of the key elements of the topology optimization process along with necessary objective and constraint functions are presented. Readers familiar with the topic may skip this chapter. The research with respect to the material interpolation mthods used for the optimization of dynamic properties is presented in Chapter 3. The method developed for mode tracking and dealing with multiplicity is described in Chapter 4. In Chapter 5, some existing and new methods to converge to a discrete design are compared, in the setting of eigenfrequency optimization. The knowledge obtained from these research sections is applied on the cases and the results of these are presented in Chapter 6. The overall results from all of these chapters are discussed in Chapter 7. Lastly, the conclusions and recommendations are presented in Chapter 8.

2

Topology optimization

This chapter introduces the topology optimization method for structural design. The basic methods of structural optimization are introduced and briefly compared. Some details with regards to the finite element models are shown along with a brief description of the optimization method. The objective and constraint functions used in this thesis are shown along with some basic filtering techniques.

2.1. Structural optimization methods

Structural optimization always plays a role in the design process, whether it's used to reduce the cost, increase performance or reduce the amount of material used. This usually goes hand in hand with analysis and simulations in an iterative design process. However the responsibility of improving the design at each iteration is shifting from the engineer towards a computer algorithm.

The types of structural optimization can be classified in three different categories: size, shape and topology optimization. The design variables of size optimization are usually the dimensions of cross-sections of trusses and frame members or the thickness of plates. When using shape optimization the design variables are used to adjust the material boundaries, for example the shapes of holes or locations of truss members. Topology optimization has even more design freedom and is able to create vastly different structures, in this case the design variable is the material density field. The different categories are illustrated in Figure 2.1.
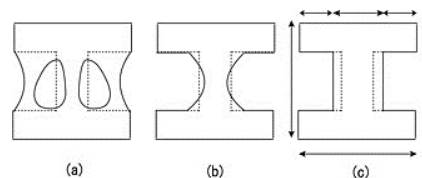
Figure 2.1: Three categories of structural optimization. a) Topology optimization, b) Shape optimization, c) Size optimization. Image source:[Kato et al., 2014]

The topology optimization method is able to produce designs with much more freedom compared to the size and shape optimization methods. Where the size and shape optimization both require a starting point the topology optimization method can start with a blank sheet. Looking at Figure 2.1 this also becomes clear, the result from the topology optimization method is much more complex compared to the shape and size optimization.The complexity of the result is sometimes a disadvantage of the method as not all design are feasible in practice. The freedom of the optimization process is in this case the cause of the problem. The design freedom can be limited somewhat by applying certain objective functions, constraints or filters. This limitation allows the method to produce manufacturable designs. Within the topology optimization field several methods have been developed, the three most common methods that will be discussed shortly are:

- Density based method (SIMP)
- Level set method (LSM)
- Evolutionary structural optimization (ESO)

2.1.1. Density based method

In the density based method each element of the finite element model is assigned a design density value. The mechanical properties of the finite element model are dependent on these design density values. All of these design density values can be adjusted by the optimizer to find the best solution. Usually the scaling is continuous from a design density value of 0 (void) to 1 (solid), for example the elasticity tensor:

$$E_{ijkl}(X) = \rho(X)\bar{E}_{ijkl} \qquad 0 \leq \rho \leq 1 \tag{2.1}$$

Here $\bar{E}_{ijkl}$ is the elasticity tensor and ρ is the design density value which is a function of the location X within the model. Other material properties are scaled in a similar way. The continuous scaling allows the use of gradient-based optimization algorithms which greatly improve the convergence rate.

A downside to this linear scaling is that intermediate densities could be optimal for the objective function. Usually a discrete design containing only solid or void elements is preferred from a manufacturers point of view. Therefore material interpolation methods with penalization of these intermediate densities have been introduced by Bendsoe and Kikuchi [Bendsøe and Kikuchi, 1988; Bendsøe, 1989]. This method of penalizing intermediate densities is known as solid isotropic material with penalization (SIMP) introduced by Rozvany et al. [Rozvany et al., 1992]. An example of penalizing the elasticity with a power-law with penalization factor p:

$$E_{ijkl}(X) = \rho(X)^p \bar{E}_{ijkl} \qquad 0 \leq \rho \leq 1 \tag{2.2}$$

The sensitivities with respect to the objective function can usually be found analytically since there is a design variable linked to every finite element. At every iteration the design density values are updated by the optimization algorithm according to the sensitivity information to converge towards an optimal solution.

2.1.2. Evolutionary structural optimization

The ESO method, introduced by Xie and Steven[Xie and Steven, 1993], is different from the density based approach. It uses discrete design density values for each element which changes the optimization to a discrete optimization problem.

It is based on the idea that the optimal solution can be obtained by slowly removing (setting the design density value to 0) the least effective elements from the design. At every iteration the solution from the finite element model can be used to determine the effectiveness of every element. The least effective elements which are below a rejection criterion are then removed until an optimal solution is found. This is also immediately the downside of the method, it is only removing elements. The bi-directional evolutionary structural optimization (BESO) has been introduced by Yang et al. [Yang et al., 1999], which uses a similar method to determine the effectiveness but is also able to add (setting the design density value to 1) elements which were removed in earlier iterations. The process of this optimization method is not very smooth due to the discrete design variables, this could be problematic when strict constraints are applied. At the same time it is also a computationally expensive method to use as it requires more iterations than the density based approach.

2.1.3. Level set method

The LSM method uses a very different approach compared to the previous methods. It uses moving boundaries as material interfaces between solid and void material. The mathematical derivation of the moving boundary problem was introduced by Osher and Sethian [Osher and Sethian, 1988], and applied to topology optimization by Wang et al. [Wang et al., 2003].

The material boundaries are constructed by summing all of the so called level set functions. These simple functions are geometrically defined in 3 or 4 dimensions and cover a small portion of the design domain. By using multiple functions the complete domain is filled. The material boundary lies at an isoline of the sum of all functions as shown in Figure 2.2. Each of the level set functions has a design variable associated to it to control it's magnitude which in turn changes the location of the isoline. This

allows for complex structures without defining very complicated curves for the boundaries.
In this case the finite element material densities are not directly linked to the design variables anymore. The total number of design variables is dependent on the number of level set functions and not on the number of finite elements. This is a benefit of the method as its computational cost does not scale with the mesh used. However the results of this method are sensitive to the initial design.
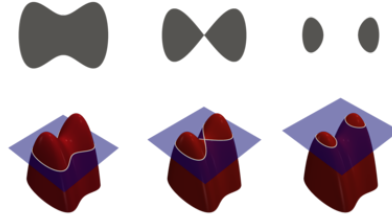
Figure 2.2: An illustration of the level set method, the geometry resulting from the level set functions is shown on the top row.

2.2. Selected optimization method

All of the presented method are suitable to optimize the dynamic performance of a structure. However in most literature on the optimization of eigenvalues, the density based method is used for its versatility and design freedom. The design freedom has an impact on the performance that could be achieved. At the same time this is also the most mature method and extensive literature is available on the method. For these reasons the density based approach will be further used in this thesis.

2.3. Finite element model

The finite element model is one of the key features of topology optimization, especially in the density based approach. Throughout this thesis there will be examples shown from different models. The finite element models that will be used consist of 2D solid elements (8 DOFS ANSYS PLANE182), 3D solid elements (24 DOFS ANSYS SOLID185) and 3D shell elements (24 DOFS ANSYS SHELL181). As the focus is on the formulations, simple rectangular meshes will be used, some examples of the same geometry are shown in Figure 2.3.

(a) 3D solid mesh   **(b)** 2D Quad mesh   **(c)** 3D Shell mesh
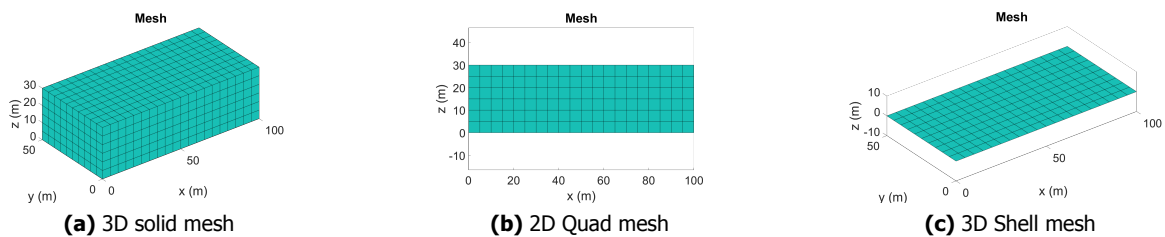
Figure 2.3: Finite element meshes using different element types of the same rectangular bar 100x50x30 m.

ANSYS APDL is used to find the mass and stiffness matrices of these elements, more details about this procedure is found in Section A.1. The material properties and size of one element is sent to ANSYS and the matrices are returned. This ensures correct computation of the mass and stiffness matrices and will serve as the basic building block of the full model. This way the ANSYS procedure has to be executed only once throughout the whole process. The mass and stiffness matrix of the full model is then built in MATLAB using the basic element, the material interpolation functions and the design densities. The eigenvalues and mode shapes of the full model are obtained using MATLAB's built in solver for a subset of eigenvalues.

2.4. Optimization algorithm

At every iteration in the topology optimization process the design variables need to be updated. This is where the method of moving asymptotes (MMA), developed by Svanberg [Svanberg, 1987], comes in. The MMA method is a type of non-linear programming suitable for gradient-based structural optimization. At every iteration a strictly convex subproblem is generated and solved. The asymptotes control the approximation of the subproblem which stabilizes the subproblem and speed up convergence.

2.5. Objective functions

The objective of the topology optimization method can be almost anything as long as it can be defined mathematically and sensitivity information with respect to the design variables is available. The objective functions required for the case studies are based on the eigenfrequencies of a structure. Therefore two different methods are presented, the optimization of a single eigenvalue and the mean eigenvalue objective. The common formulation of a minimization problem for the structural optimization process is shown below. This formulation was first introduced by Taylor and Bendsoe [Taylor and Bendsøe, 1984].

$$Min \qquad f_{obj}(\rho) \tag{2.3}$$

Subject to:

$$(K - \lambda_j M)\phi_j = 0, \qquad j = 1, ..., N \tag{2.4}$$

$$\phi_j^T M \phi_k = \delta_{jk}, \qquad j, k = 1, ..., N \tag{2.5}$$

$$0 < \rho_{min} < \rho_e \leq 1, \qquad e = 1, ..., Ne \tag{2.6}$$

$$\lambda_j = \omega_j^2 \tag{2.7}$$

Equations 2.4 and 2.5 are the state equations of the eigenvalue problem and the mass orthogonality of eigenmodes respectively. The constraint shown in Equation 2.6 is a constraint on the design density variable. A lower bound is set on the design variable in order to avoid numerical singularities.

2.5.1. Single eigenvalue objective

The objective function for the maximization of the nth eigenfrequency of a structure is formulated as a simple eigenvalue optimization problem as shown by Ma et al. [Ma et al., 1993; 1994; 1995]. The sensitivity analysis for this objective function is presented in Section C.1.

$$f_{obj}(\rho) = -\lambda_n \qquad \lambda_n = \omega_n^2 \tag{2.8}$$

2.5.2. Mean eigenvalue objective

In most physical systems multiple eigenfrequencies are usually of interest which is why there is a need to optimize multiple of these simultaneously. The versatile mean-eigenvalue formulation introduced by Ma et al. [Ma et al., 1994] can be used to optimize multiple weighted eigenvalues simultaneously. This can be seen as the sum of reciprocal eigenvalues, meaning the lowest eigenvalue will have the biggest contribution to the objective value. The sensitivity analysis for this objective function is presented in Section C.2.

$$f_{obj}(\rho) = -\lambda^* \qquad \lambda^* = \lambda_t + \sum_{i=1}^{m} w_i \left(\sum_{i=1}^{m} \frac{w_i}{\lambda_i - \lambda_t} \right)^{-1} \tag{2.9}$$

In which λ_i is the ith eigenvalue of the system and w_i are the corresponding weighting factors. The parameter λ_t determines around which value is being optimized as eigenvalues close to λ_t have the largest contribution to the objective value λ^* when all the weighting factors are equal. In that case the eigenvalue closest to λ_t will be optimized.
If $\lambda_t = 0$, then the smallest eigenvalue will have the largest contribution to the mean-eigenvalue, so the lowest eigenvalue of the structure will be optimized. If $m = 1$ the formulation reduces to the optimization of a single eigenvalue λ_i as in Equation 2.8. A more generalized approach was also presented in the same paper and is an extended form of the mean-eigenvalue formulation:

$$\lambda^* = \begin{cases} \lambda_t^* + \left(\dfrac{1}{\sum_{i=1}^{m} w_i} \displaystyle\sum_{i=1}^{m} w_i (\lambda_i - \lambda_{t_i})^n \right)^{1/n} & \text{for} \quad n = \pm 1, \pm 2, \dots n \neq 0 \\[2em] \lambda_t^* + \exp\left(\dfrac{1}{\sum_{i=1}^{m} w_i} \displaystyle\sum_{i=1}^{m} w_i \ln |\lambda_i - \lambda_{t_i}| \right) & \text{for} \quad n = 0 \end{cases} \tag{2.10}$$

In which n is a given power and λ_{t_i} are the parameters for the ith eigenvalues. The behavior of the mean eigenvalue objective can be summarized as follows:

- For $n \geq 0$ the eigenvalue furthest from λ_{t_i} will have the largest contribution to the mean eigenvalue.
- For $n < 0$ the eigenvalue closest to λ_{t_i} will have the largest contribution to the mean eigenvalue.
- For n as an odd number, the increase of any eigenvalue will result in the increase of the mean eigenvalue.
- For n as an even number the increase of distance $\lambda_i - \lambda_{t_i}$ will result in an increase of the mean eigenvalue.

2.6. Constraint functions

The MMA algorithm is only able to deal with inequalities, so in order to constrain a property such as the ratio of two eigenfrequencies, two constraints are required. These constraint enforce an upper and lower limit on the property. In order to ensure equal scaling of all constraint values, a normalization is applied on all constraints. Each constraint has a cost value linked to it, this cost value will determine importance of the constraint to the optimizer. The most costly constraint will be satisfied first.

2.6.1. Volume constraint

To constrain the amount of material used in the design domain, a volume constraint can be applied. This is the sum of the volume of each element v_e multiplied by its design density value ρ_e. A relative upper

limit of V_{max} and a lower limit of V_{min} can be specified. The sensitivity analysis for these constraints is presented in Section D.1.

$$\frac{\sum_{e=1}^{n} \rho_e v_e}{V \cdot V_{max}} - 1 < 0 \qquad 1 - \frac{\sum_{e=1}^{n} \rho_e v_e}{V \cdot V_{min}} < 0 \qquad\qquad (2.11)$$

2.6.2. Eigenfrequency ratio constraint

The eigenfrequency ratio r_{ij} as defined in Equation 2.12 can be constrained by using the two constraints shown in 2.13. The sensitivity analysis for these constraints is presented in Section D.2.

$$\frac{\lambda_i}{\lambda_j} = r_{ij} \qquad \frac{\omega_i^2}{\omega_j^2} = r_{ij}^2 \qquad\qquad (2.12)$$

$$\frac{\lambda_i}{\lambda_j r_{ij}} - 1 < 0 \qquad 1 - \frac{\lambda_i}{\lambda_j r_{ij}} < 0 \qquad\qquad (2.13)$$

2.6.3. Static compliance constraint

When optimizing for eigenfrequencies of a structure the result in some cases can be a relatively weak structure when loaded statically. This is why in some cases a constraint on the maximum static compliance of a structure is applied. A load vector F_i can be used to find the displacement vector u using Equation 2.14. The compliance C_i corresponding to load vector F_i can be found using Equation 2.15. A maximum compliance C_{max} can be enforced using the constraint shown in 2.16. The sensitivity analysis for this constraints is presented in Section D.3.

$$Ku = F_i \qquad\qquad (2.14)$$

$$C_i = F_i^T u \qquad C_i = u^T K u \qquad\qquad (2.15)$$

$$\frac{C_i}{C_{max}} - 1 < 0 \qquad\qquad (2.16)$$

2.6.4. Stiffness constraint

Instead of the compliance of the total structure with respect to a load vector, the stiffness of a structure in a certain direction at a certain location can be constrained in a similar way. By using a load vector F_i with only a unit force on the DOF of interest, resulting in displacement u_i of this DOF. The stiffness k_i of this DOF can then be evaluated using Equation 2.18. A constraint on the minimum stiffness of DOF i can be applied using the constraint in 2.19. The sensitivity analysis for this constraint is presented in Section D.4.

$$u_i = F_i^t u \qquad u_i = u^T K u \qquad\qquad (2.17)$$

$$k_i = \frac{1}{u_i} \qquad\qquad (2.18)$$

$$\frac{k_{min}}{k_i} - 1 < 0 \qquad\qquad (2.19)$$

2.7. Filtering

Density-based optimization suffers from a few problems regarding manufacturability and mesh independence, which is why filters are applied to solve these issues. Many different filtering methods have been presented but the perfect filter for topology optimization does not yet exist. Some of the most common filters are presented here and will be applied throughout this thesis. The common feature that these filters have is that they work as a weighted average on the neighborhood of an element. This neighborhood with radius R, defined in Equation 2.20, contains all elements with their center x_i located within the radius. This radius in some cases is also directly linked to the minimum feature size that is possible within the design domain.

$$N_e = (i|\ ||x_i - x_e|| \leq R) \tag{2.20}$$

2.7.1. Density filtering

In the case of density filtering, the physical density of an element is a function of the design density values within the neighborhood of that element as shown in Equation 2.21. Before building the mass and stiffness matrix the design densities are transformed to the physical densities, consequently the sensitivity information computed on the physical densities has to be modified to take this into account.

$$\tilde{\rho}_e = \tilde{\rho}_e(\rho_i \in N_e) \tag{2.21}$$

Basic density filter

The basic density filter introduced by Bruns and Tortorelli [Bruns and Tortorelli, 2001] is a simple weighted average on the neighborhood. The filter equation is shown in Equation 2.22 in which $w(x_i)$ is a cone shaped linear decaying weighting function as in Equation 2.23. As mentioned before v_i is the element volume in case varying mesh sizes are used.

$$\tilde{\rho}_e = \frac{\sum_{i \in N_e} w(x_i) v_i \rho_i}{\sum_{i \in N_e} w(x_i) v_i} \tag{2.22}$$

$$w(x_i) = R - ||x_i - x_e|| \tag{2.23}$$

Heaviside filter

A downside of the basic filtering scheme is the blurring of the edges, Guest et al. [Guest et al., 2004] introduced the density filter with a Heaviside step transformation applied after the basic density filter. This step function shown in Equation 2.24 will act as a step function when $\tilde{\rho}_e > 0$, the magnitude of this effect is controlled by the β parameter as shown in Figure 2.4. When $\beta = 0$ it acts like the basic density filter and in the case of $\beta = \infty$ it acts like a maximum of the neighborhood, setting the center element to 1 if any of the elements within the neighborhood is larger then zero.

$$\tilde{\rho}_e = 1 - e^{-\beta \tilde{\rho}_e} + \tilde{\rho}_e e^{-\beta} \tag{2.24}$$

Modified heaviside filter

In a similar approach a modified version of the Heaviside filter was introduced [Xu and Yuan, 2004]. But this version works as a minimum on the neighborhood. For a high value of the β it will return zero for $\tilde{\rho}_e < 1$ and one for $\tilde{\rho}_e = 1$ as shown in Figure 2.4. The center element in this case will only have a value of one if all elements in the neighborhood have a value of 1. The resulting topology from this filter shows no features smaller then the filter radius.

$$\tilde{\rho}_e = e^{-\beta(1 - \tilde{\rho}_e)} - (1 - \tilde{\rho}_e)e^{-\beta} \tag{2.25}$$
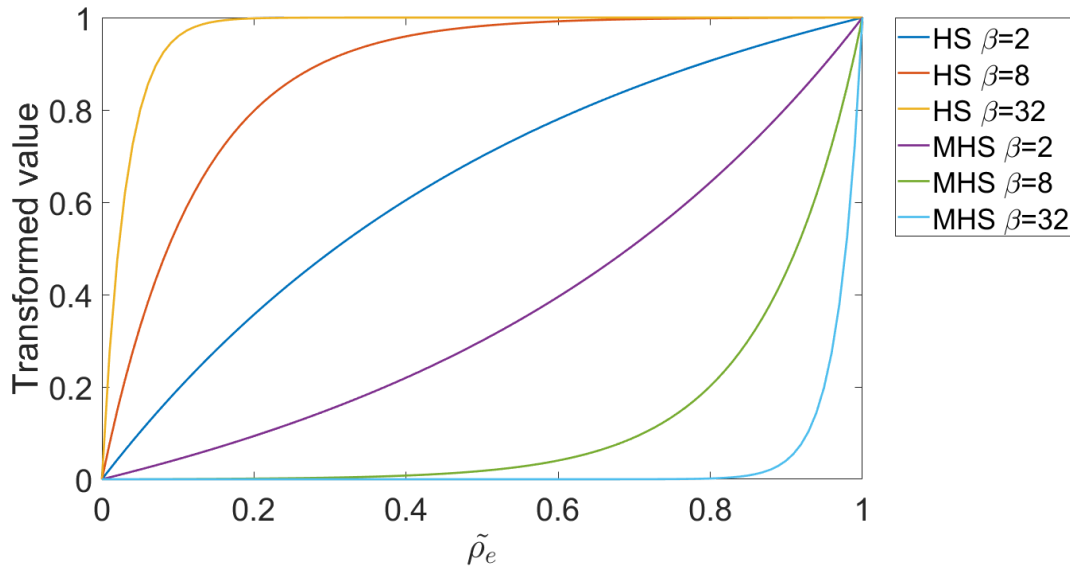
Figure 2.4: Heaviside (HS) and modified heaviside (MHS) transformation for different values of beta.

2.7.2. Sensitivity filtering

The filtering of the sensitivity information was introduced by Sigmund [Sigmund, 1994] as a heuristic measure to solve the checker boarding problem described by Sigmund and Petersson [Sigmund and Petersson, 1998], and ensure a mesh-independent result. It works similar to the density method by using a weighted average of the sensitivity information in the neighborhood of every element. The filter is shown in Equation 2.26. In a more recent paper by Sigmund and Maute [Sigmund and Maute, 2012] the method is analyzed from a continuum mechanics perspective and a link is shown with nonlocal elasticity concepts.

$$\frac{\partial f_{obj}}{\partial \rho_e} = \frac{\sum_{i \in N_e} w(x_i) \rho_i \frac{\partial f_{obj}}{\partial \rho_i}}{\rho_e \sum_{i \in N_e} w(x_i)} \tag{2.26}$$

3

Material interpolation methods

In this chapter some of the problems related to the material interpolation model will be discussed. A detailed investigation into the parameters shows the origin of these issues. These results are used to determine the requirements for a material interpolation method suited for topology optimization of constrained eigenvalue problems. New and existing models are compared with these requirements and conclusions are drawn.

3.1. Problem analysis

The purpose of the material interpolation method is to provide a link between the physical properties and the design density variables. At both ends of the design density range it is clear that it should represent the solid material and void material. However, elements that are completely void with no stiffness or mass would not work numerically due to singularity problems, which is why there is always a minimum value on the design density. The effect of this void material is investigated further in Section 3.1.2.

One of the issues that originates from the choice of material interpolation method is the occurrence of local eigenmodes within the first few eigenfrequencies of a structure. This can be problematic to the optimization process if handled incorrectly. These local eigenmodes are affected by several parameters of the material interpolation methods which are investigated in Section 3.1.1.

3.1.1. Local eigenmodes

When optimizing a structure the final result will most likely be a structure with some solid and void areas. In these void areas the so called local eigenmodes can occur, two examples of local eigenmodes are shown in Figure 3.1. The design densities in the void areas are so low compared to the solid areas that the solid areas start to act like a fixed boundary condition. These eigenmodes are always present in a structure with void areas but their frequency can be drastically reduced by the penalization of the material interpolation functions.

Local eigenmodes and methods to avoid them have been investigated previously by Pedersen [Pedersen, 2000] in case of the SIMP method. A local eigenmode is characterized by the vibration of relatively high mass and low stiffness elements due to the minimum design density, which makes them occur at lower frequencies. The heuristic measures taken in literature focus on improving the relatively poor ratio between mass and stiffness for void elements.

In the optimization process these local eigenmodes can be problematic for two reasons. Since the optimization is normally focused on the first few structural modes, only these are of interest. When many local eigenmodes are present within the frequency range of the first few structural modes, the number of modes that needs to be computed increases, which increases the computation time for every iteration. Besides that it is also necessary to track the order of the structural modes to ensure correct computation of the objective and constraint values and corresponding gradients. Having many local eigenmodes present could be problematic for the mode tracking.
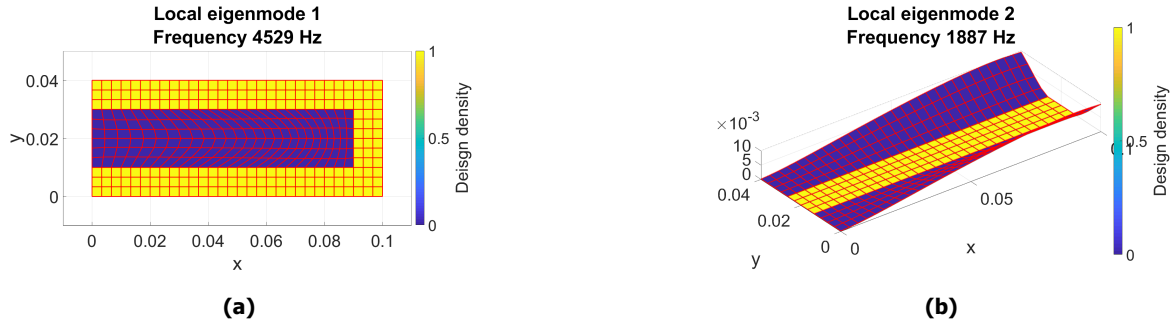
Figure 3.1: Examples of some local eigenmodes with different element types a) Local eigenmode with 2D solid element mesh b) Local eigenmode with 3D shell element mesh

Effect of penalization on local eigenmode frequency

The SIMP method will be used to investigate the effect of penalization factors on the local eigenfrequencies. The equations to obtain the stiffness and mass matrix of the full model are shown in Equation 3.1, in which k_e and m_e are the element matrices obtained from ANSYS.

As mentioned before the penalization factors of the SIMP method have an effect on the frequency of the local eigenmodes. To investigate this effect, the model shown in Figure 3.1a has been used to sweep the penalization factors of the mass (P_m) and stiffness (P_k).

The effect of both penalizations factors on the relative change of the local eigenfrequency is shown in Figure 3.2. Two different values of the void material design density ρ_{min} have been used. While sweeping one of the penalization factors, the other is set to 1.

$$K(\rho) = \sum_{e=1}^{n} \rho_e^{P_k} \cdot k_e \qquad M(\rho) = \sum_{e=1}^{n} \rho_e^{P_m} \cdot m_e \qquad \rho_{min} \leq \rho \leq 1 \tag{3.1}$$

(a) $P_k = 1 - 3 \; P_m = 1$

(b) $P_k = 1 \; P_m = 1 - 3$
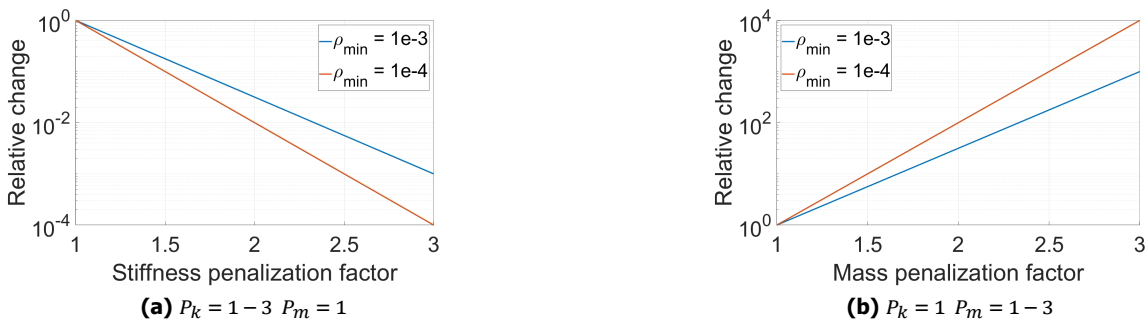
Figure 3.2: Effect of the a) stiffness and b) mass penalization factors on the local eigenfrequencies

The graphs show that there is a clear relationship between the penalizations factors and the minimum design density. This relationship can be found using the equation for a resonance frequency as shown in Equation 3.2. Replacing the mass and stiffness by the penalized value at minimum design density the relationship shown in Equation 3.3 is found. In this equation ω_0 is the frequency of the local eigenmode with penalization factors of 1.

$$\omega = \sqrt{\frac{K}{M}} \tag{3.2}$$

$$\frac{\omega_{local}}{\omega_0} = \sqrt{\frac{\rho_{min}^{P_k}}{\rho_{min}^{P_m}}} \tag{3.3}$$

Effect of k/m ratio on local eigenmode frequency

The general conclusion that can be taken from Equation 3.3 is that the local eigenfrequencies are dependent on the ratio of the mass and stiffness penalization equations. This is not only true for the SIMP method but for all material interpolation methods. Equation 3.3 can be converted to a more general equation for all methods as shown in Equation 3.4, in which f_k is the stiffness interpolation function and f_m is the mass interpolation function. With this equation the effect of different material interpolation methods on the local eigenfrequencies can be predicted.

$$\frac{\omega_{local}}{\omega_0} = \sqrt{\frac{f_k(\rho_{min})}{f_m(\rho_{min})}} \tag{3.4}$$

Eigenmode properties

From literature it is clear that the local eigenmodes are often occurring and heuristic measures are taken to avoid them. But these measures are not preferred as they introduce non-linearities in the material interpolation functions. A better solution is detecting them and removing them from the eigenvalue analysis results.

In order to detect the local eigenmodes, a selection criterion is needed. Several properties of an eigenmode can be computed such as the potential and kinetic energy, modal mass and modal stiffness. It is relevant here to consider the different ways eigenmodes are scaled, the properties change depending on the scaling used. The mode shapes obtained from the eigenvalue analysis are all scaled according to the mass normalization shown in Equation 3.5, which means that their individual amplitude is different. The model shown in Figure 3.1a is used to analyze the eigenmode properties. The norm of eigenmodes 1 to 50 is shown in Figure 3.3, eigenmodes 1 to 21 are structural modes and all show a similar amplitude. Some of the higher modes show a much larger amplitude, these are the local eigenmodes. This is due to the fact that the vibrating elements of a local eigenmode have a low mass associated with them, requiring a high amplitude to satisfy the mass normalization equation. A distinction can already be made by observing the maximum amplitude but this is very unpredictable and model dependent.

$$\phi_i^T M \phi_j = \delta_{ij} \tag{3.5}$$
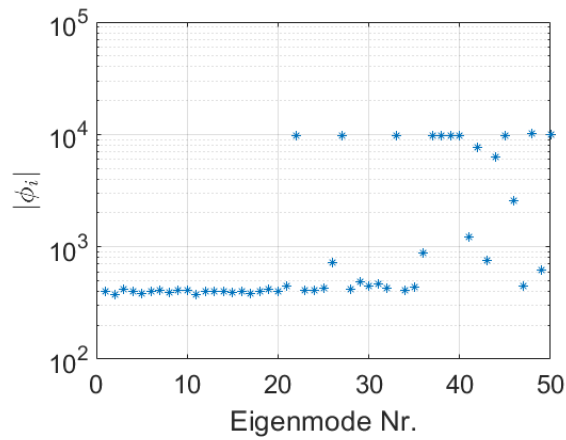
Figure 3.3: Vector norm of eigenmodes 1-50.

Several more predictable properties that could be analyzed and their expected behavior for a solid model are shown below.

- Potential energy. Energy stored in the deformation shape of the mode. Higher order modes usually have more strain in their mode shapes so the potential energy should increase with the eigenfrequency.

$$E_{pi} = 0.5 \cdot \phi_i^T K \phi_i \tag{3.6}$$

- Kinetic energy. Energy associated with the vibrating mass of the eigenmode. The vibrating mass generally does not change, so the kinetic energy increases with increasing eigenfrequency.

$$E_{ki} = 0.5 \cdot \omega_i^2 \phi_i^T M \phi_i \tag{3.7}$$

- Modal stiffness. Substitute stiffness when decoupling the equations of motion of the full model using modal superposition. Represents the stiffness of the eigenmode. Should go up with the eigenfrequency of the modes as the higher modes have more strain in their mode shapes.

$$\gamma_i = \phi_i^T K \phi_i \tag{3.8}$$

- Modal mass. Substitute mass when decoupling the equations of motion of the full model using modal superposition. Represents the vibrating mass of the eigenmode. Should remain fairly equal for a solid model.

$$\mu_i = \phi_i^T M \phi_i \tag{3.9}$$

A different scaling is required to analyze the mode shape properties since all properties can already be computed when mass normalization is applied, by using the Rayleigh quotiënt shown in Equation 3.10.

$$\omega_i^2 = \frac{\phi_i^T K \phi_j}{\phi_i^T M \phi_j} \tag{3.10}$$

The eigenmodes can also be normalized using Equation 3.11, this equalizes the amplitudes of all eigenmodes.

$$|\phi_i| = 1 \tag{3.11}$$

The eigenmode properties for the first 50 modes of the model shown in Figure 3.1a are shown in Figure 3.4. As expected the potential and kinetic energy increase with increasing eigenfrequency, they are both equal to each other as well. This is due to the fact that energy of a harmonic vibration is continuously being exchanged between kinetic and potential energy.

The modal stiffness is simply double the potential energy so it shows the same behavior. The modal mass also shows the expected behavior of remaining relatively constant with increasing eigenfrequencies as the mass of the model remains the same.
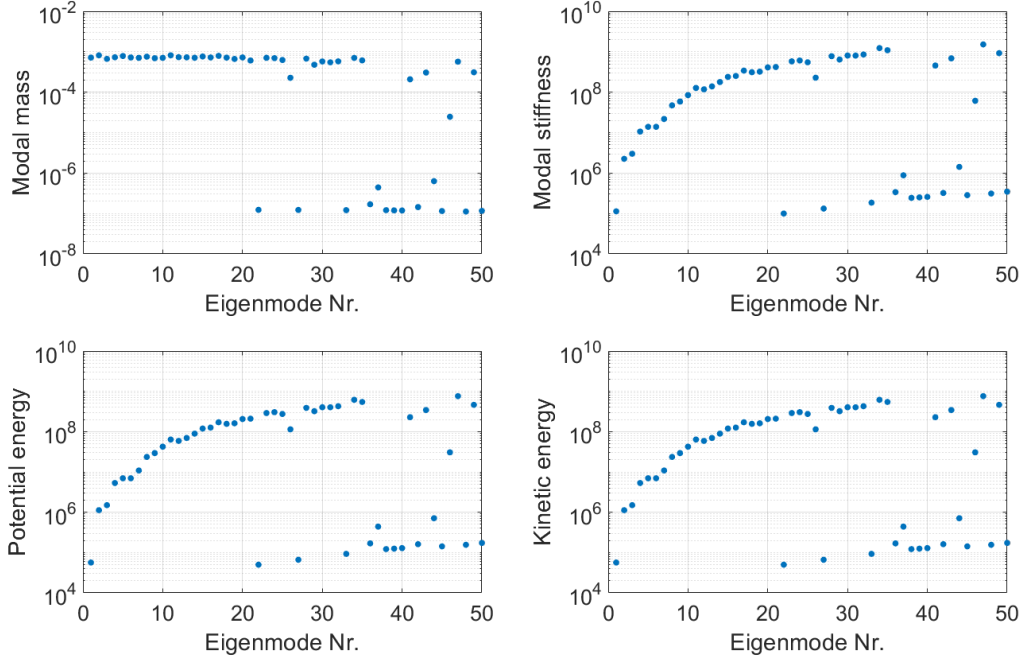
Figure 3.4: Mode shape properties

Method to detect local eigenmodes

From the modal mass in Figure 3.4 it becomes clear that after mode number 21 the local eigenmodes start showing up. In fact the first local eigenmode shown in Figure 3.1a corresponds to the 22nd mode. This shows that the modal mass can be used as an indicator of local eigenmodes.

The values of the modal mass can be predicted by simple reasoning. A model with a single mass of 1 kg and a single degree of freedom will have a normalized mode shape with a unit displacement value for that degree of freedom. Filling in Equation 3.9 then gives 1 kg m^2 as a value for the modal mass. Due to the fact that the modes are normalized with Equation 3.11, the resulting modal mass value is equal to the single mass.

Now going back to the finite element model, this still holds true. But instead of the mass of the full model, the average mass of all vibrating elements is returned. More specifically the weighted average of moving mass, weighted by their amplitude in the mode shape. A structural eigenmode should have a modal mass value near the value predicted by Equation 3.12 while a local eigenmode should have a value near the mass of a void element predicted by Equation 3.13. In this equation $f_m(\rho)$ is the mass interpolation function and v_i is the element volume.

$$\mu_{structural} = \frac{\sum_{i=1}^{n} f_m(\rho_i)v_i}{n} \cdot \rho_{material} \tag{3.12}$$

$$\mu_{local} = f_m(\rho_{min}) \cdot \frac{\sum_{i=1}^{n} v_i}{n} \cdot \rho_{material} \tag{3.13}$$

The modal mass values along with the predicted values are shown in Figure 3.5. The values computed using the equations are a good approximation of the modal mass for both structural and local eigenmodes. A clear separation can be made between structural and local eigenmodes. There are however some exceptions where the mode shapes show interaction between the solid and void parts and it is unclear whether these modes should be classified as structural or local.
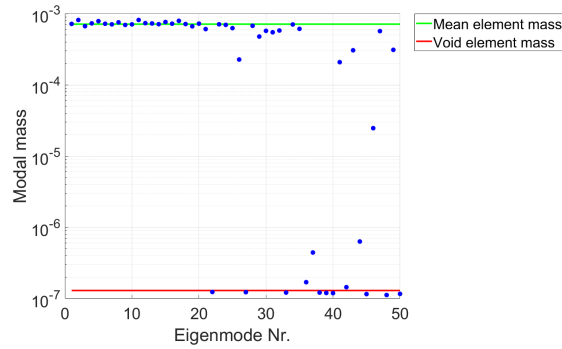
Figure 3.5: Modal mass values with predicted mean and void values

Methods to prevent local eigenmodes

Several methods to prevent the local eigenmodes from occurring at low frequency have been presented in literature.

- Applying two different interpolation functions for the mass, introduced by Pedersen [Pedersen, 2000]. The linear mass interpolation function is used between a threshold value and 1, while an interpolation function with a higher penalization factor is used between zero and the threshold value. This results in an increase of the eigenfrequency of the local eigenmodes. This could be done with either a C^0 or C^1 continuity on the interpolation function, introduced by Du and Olhoff [Du and Olhoff, 2007].

- Changing the interpolation domain, introduced by Pedersen [Pedersen, 2000]. By changing the interpolation domain from $[0, 1]$ to $[\epsilon, 1]$, a minimum value for both mass and stiffness can be defined. The ratio of these values determines the frequency drop of the local eigenmodes.

- Reducing the mass to zero, introduced by Tcherniak [Tcherniak, 2002]. By reducing the mass of an element below a certain threshold on the design density to a small value, the local eigenfrequencies are artificially increased.

All these methods have one thing in common, they all aim to improve the ratio of the mass and stiffness interpolation functions at the lower ranges of the design density domain. The problem however lies at the choice of the material interpolation functions, and that is also where it could be fixed. Some of these methods also introduce a discontinuity in the mass interpolation function, discontinuities are rarely a good idea in gradient-based optimization.

3.1.2. Effects of void material

The final result of a topology optimization problem will in many cases have areas which are completely void. Numerically there always needs to be a minimum positive value on the design density to prevent singularity of the stiffness matrix. Void elements physically represent the absence of material, therefore this void material should not be affecting the structural integrity of the model.

In the case of dynamics the structural eigenfrequencies should remain equal with or without void material present. The model shown in Figure 3.1a and the SIMP material interpolation method will be used to investigate the effect of the void material on the structural eigenfrequencies.

Effect of void material on structural mode eigenfrequencies

The effect that the void material has on the structural eigenfrequencies is dependent of a few factors. The amount and location of the void material present has an impact on the magnitude of this effect. The material interpolation functions used also have an impact on this effect, this determines whether the void material represents a relatively weak and heavy ($\frac{P_k}{P_m} = 2$) or a stiff and light material ($\frac{P_k}{P_m} = 0.5$).

The resulting relative effect on the first structural eigenfrequency is shown in Figure 3.6. These results are obtained using the model shown in Figure 3.1a, the void area in this model is roughly 45% of the total structure. In this case ω_r is the eigenfrequency at the lowest value of the minimum design density. The baseline with penalization factors of 1 shows that in this case the minimum design density
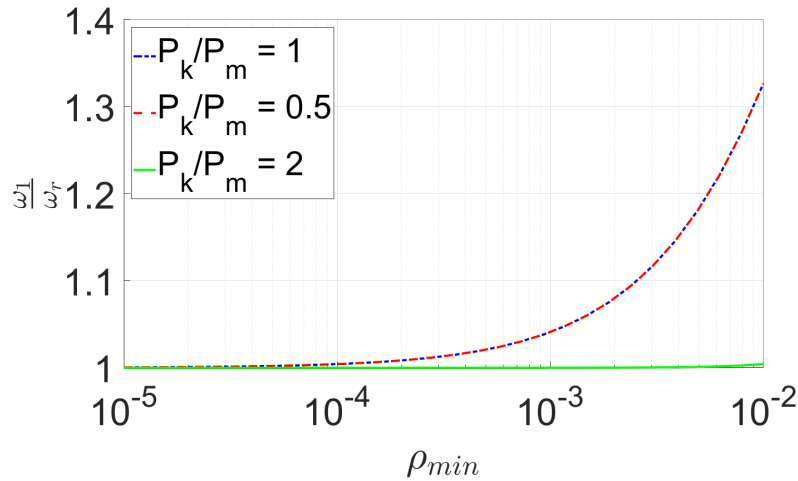
Figure 3.6: Effect of minimum design density value on a structural eigenfrequency.

value has quite an impact on the structural eigenfrequency. Applying a mass penalization factor of 2 increases this effect very slightly as the same amount of stiffness is added by the void material but less mass is added. Applying a stiffness penalization factor of 2 actually reduces the effect significantly as this reduces the added stiffness of the void material drastically. Therefore the effect of void material on the structural eigenfrequencies is mostly related to the added stiffness of void material.

Effect of minimum design density on local mode eigenfrequency

With the relationship found in Equation 3.4, the effect of the minimum design density value on the local eigenfrequencies can be predicted. Three cases can be determined.

- Equal penalization. As long as the ratio of the penalization functions is equal to 1, the eigenfrequencies of the local modes remain unchanged, independently of the minimum design density.
- Higher stiffness penalization. This results in a decrease of the local eigenfrequencies, increasing the value of the minimum design density will increase the local eigenfrequencies.
- Higher mass penalization. The local eigenfrequencies are increased when this is the case, reducing the value of the minimum design density will increase the local eigenfrequencies.

3.2. Requirements for an ideal material interpolation method

The ideal material interpolation method for optimization of eigenvalues does not yet exist, however some requirements for an ideal interpolation method can be defined as follows.

- Penalize intermediate densities. The main goal of the material interpolation functions is to provide a basis for the gradient-based optimization method to converge to manufacturable solid-void designs. By applying some sort of penalization the effectiveness of intermediate density elements is reduced, forcing the design density to go to the more efficient solid or void values.
- Bidirectional. Throughout the optimization process some elements might be forced towards void but in a later stage of the process they might improve the objective function by returning to solid. As the optimization process is gradient-based, the gradient values should never become zero throughout the design density range.
- Reduce the effect of void material. Whenever there are strict requirements on the structural eigenfrequencies, the resulting geometry should have the expected values in reality. Which is why the effect of void material should be reduced as much as possible.
- Reduce the presence of local modes. Having local eigenmodes occur at lower frequencies can be problematic to the optimization process. Therefore it would be preferred to solve the problem at the source.

3.3. Material interpolation functions

Most of the material interpolation methods presented in literature are used for the optimization of static problems. Therefore most formulations are presented with only the stiffness interpolation. In the case of the optimization of eigenvalue problems, a linear interpolation or power law interpolation is commonly used for the mass interpolation. The mass and stiffness of the full FEM model are assembled using the material interpolation functions and Equation 3.14. A few of the most commonly used material interpolation functions will we discussed next.

$$K(\rho) = \sum_{e=1}^{n} f_k(\rho_e) \cdot k_e \qquad M(\rho) = \sum_{e=1}^{n} f_m(\rho_e) \cdot m_e \tag{3.14}$$

3.3.1. Existing methods

Power law

One of the first methods presented in literature is the power law method presented by Rozvany et al. [Rozvany et al., 1992]. This method uses penalization factors as power on the design density value. The method was proposed for the optimization of static problems, which is why no mass interpolation function was presented. However for completeness both functions are shown in Equation 3.15 and a graphical presentation with different stiffness penalization factors is given in Figure B.1 in appendix B.

$$f_k(\rho) = \rho^{P_k} \qquad f_m(\rho) = \rho^{P_m} \qquad \rho_{min} \leq \rho \leq 1 \tag{3.15}$$

One of the first methods to improve the frequency of the local eigenmodes was presented by Pedersen [Pedersen, 2000]. This method changes the domain of the interpolation function to always have the same minimum value ϵ which allows for tuning of the local eigenfrequencies. The value of the minimum design density can in this case also be set to zero. A graphical presentation with different stiffness penalization factors is given in Figure B.2.

$$f_k(\rho) = \epsilon + \rho^{P_k} \cdot (1-\epsilon) \qquad f_m(\rho) = \epsilon + \rho^{P_m} \cdot (1-\epsilon) \qquad 0 \leq \rho \leq 1 \tag{3.16}$$

RAMP

An alternative to the power law interpolation functions was introduced by Stolpe and Svanberg [Stolpe and Svanberg, 2001], which is now well known as the RAMP method. This method has been developed with some theoretical propositions, which should make compliance optimization a concave function of the design densities. By increasing the penalization factor, the probability of obtaining a discrete design should increase. A graphical presentation with different stiffness penalization factors is given in Figure B.3.

$$f_k(\rho) = \frac{\rho}{1 + P_k(1-\rho)} \qquad \rho_{min} \leq \rho \leq 1 \tag{3.17}$$

Polynomial

A different take on the local eigenmode problem with a new interpolation method was presented by Zhu et al. [Zhu et al., 2010]. This method combines linear and power law interpolation with a ratio r. This ratio can be used to tune the local eigenfrequencies somewhat. A graphical presentation with different stiffness penalization factors for a ratio of $r = 0.5$ is given in Figure B.4.

$$f_k(\rho) = r\rho^{P_k} + (1-r)\rho \qquad \rho_{min} \leq \rho \leq 1 \tag{3.18}$$

SINH

Another alternative to the power law interpolation functions was introduced by Bruns [Bruns, 2005]. This method, using hyperbolic sine functions, shows a similar behavior as the RAMP method for the stiffness penalization. A second interpolation function was introduced which could be used to penalize the mass of an element. When using both of these functions simultaneously the dynamic efficiency

of intermediate densities is reduced drastically. This should improve convergence towards a discrete design as a solid or void elements is most efficient in terms of dynamics. A graphical presentation with different mass and stiffness penalization factors is given in Figure B.5.

$$f_k(\rho) = \frac{\sinh(P_k\rho)}{\sinh(P_k)} \qquad f_m(\rho) = 1 - \frac{\sinh(P_m(1-\rho))}{\sinh(P_m)} \qquad \rho_{min} \le \rho \le 1 \tag{3.19}$$

3.3.2. New methods

All of the material interpolation methods presented thus far are all derived from topology optimization of compliance problems, there is a lack of material interpolation methods specifically for topology optimization of eigenvalues. To the authors knowledge there also is no comparative study on the use of different material interpolation methods for constrained topology optimization of eigenvalues.

The new interpolation methods have been developed with the knowledge obtained from this research. The goal of these new methods is to satisfy the requirements set for an ideal material interpolation method for constrained topology optimization of eigenvalue problems.

Looking at these requirements, a linear interpolation function is able to fulfill all requirements except for the penalization of intermediate densities. Therefore a linear interpolation function will be used as a basis for a new interpolation method by enhancing it with a shape function. The reason why intermediate densities can occur in the final solution of the optimization process is related to the objective sensitivity. These values are obtained using Equation 3.20 when optimizing a single eigenvalue. A detailed derivation of this equation is given in Section C.1.

$$\frac{\partial \lambda_j}{\partial \rho_e} = \phi_j^T \frac{\partial K}{\partial \rho_e} \phi_j - \phi_j^T \lambda_j \frac{\partial M}{\partial \rho_e} \phi_j \qquad = \qquad 2\left(\frac{\partial E_p}{\partial \rho_e} - \frac{\partial E_k}{\partial \rho_e}\right) \tag{3.20}$$

In the final result there are three possibilities for the design density value of an element when optimizing for a maximum eigenvalue.

- Void. The objective sensitivity of this element is negative, the added stiffness does not overcome the effect of the added mass.
- Solid. The objective sensitivity of this element is positive, the added stiffness does over the effect of the added mass.
- Intermediate. The objective sensitivity of this element is zero, both effects cancel each other out.

As the optimization is gradient-based there is no reason for the optimizer to force the intermediate densities to either void or solid. A sensitivity value of zero indicates that there is a balance between the derivative of the kinetic and potential energy for that element. So in order to change the value from zero, this balance needs to be changed. The material interpolation functions can directly influence this balance as they are used to find the derivative of the stiffness and mass with respect to the design density according to Equation 3.14.

So if the intermediate densities above a design density value of $\rho_e = 0.5$ should be forced towards solid then the derivative value of the stiffness interpolation should be higher then that of the mass interpolation. The contrary should be true for design density values below $\rho_e = 0.5$.

Full sine

The first method presented is obtained by adding a full sine wave to the derivative of the linear interpolation and integrating the resulting new derivative to obtain the new interpolation function. A graphical presentation with different mass and stiffness penalization factors is given in Figure B.6. The amplitude of this sine wave is controlled by the penalization factors. A downside of this method is that the balance of derivatives returns to 1 at both ends of the design domain which in practice turns out to force elements towards void and solid but not completely achieving a discrete design.

$$f_k(\rho) = \rho - \frac{P_k}{2\pi}(1 - \cos(2\pi\rho)) \qquad f_m(\rho) = \rho + \frac{P_m}{2\pi}(1 - \cos(2\pi\rho)) \qquad \rho_{min} \le \rho \le 1 \qquad 0 \le P_k, P_m \le 1$$
$$\tag{3.21}$$

Half cosine

A second method presented here is obtained by adding half of a cosine wave to the derivative of the linear interpolation and integrating to obtain the new interpolation function. A graphical presentation with different mass and stiffness penalization factors is given in Figure B.7. The amplitude is again controlled by the interpolation factors. With this method the penalization of intermediate density increases towards the ends of the design domain which in practice forces the elements fully towards void and solid. The downside here is however that the eigenfrequencies of the local eigenmodes decrease with increasing penalization factors. This could be prevented by applying Pedersen's method shown in Equation 3.16.

$$f_k(\rho) = \rho - \frac{P_k}{\pi} \sin(\pi\rho) \qquad f_m(\rho) = \rho + \frac{P_m}{\pi} \sin(\pi\rho) \qquad \rho_{min} \leq \rho \leq 1 \qquad 0 \leq P_k, P_m \leq 1 \qquad (3.22)$$

Linear derivative

A final method presented here is obtained by adding linear functions to the derivative of the linear interpolation and integrating to obtain the new interpolation function. A graphical presentation with different mass and stiffness penalization factors is given in Figure B.8. In this case the penalization factors control the slope of the derivative. This method behaves similarly to the half cosine method.

$$f_k = P_k\rho^2 + (1 - P_k)\rho \qquad f_m = -P_m\rho^2 + (1 + P_m)\rho \qquad \rho_{min} \leq \rho \leq 1 \qquad 0 \leq P_k, P_m \leq 1 \qquad (3.23)$$

3.4. Comparison

In order to compare the different methods presented, a benchmark case is used. A commonly used benchmark in literature is shown in Figure 3.7, which is a cantilever pinned on both sides. The objective is to optimize the fundamental eigenfrequency of the structure using 50% of the available design domain. The mesh-independency filter introduced in Section 2.7.2 is used to prevent checker-boarding patterns from occurring. To compare the performance of the final result, the topology is reevaluated using a linear interpolation method after the optimization has converged.
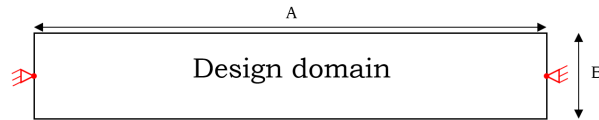
Figure 3.7: Boundary conditions and design domain dimensions for the benchmark case.

The Measure of non-discreteness introduced by Sigmund [Sigmund, 2007] is used to compare the convergence towards a fully discrete design. This measure will be 100% in the case where all elements have a design density of 0.5 and will be 0% when they are at either 0 or 1. The performance results of all methods are summarized in Table 3.1 with the corresponding topology shown in Figure 3.8.

$$M_{nd} = \frac{\sum_{e=1}^{n} 4\rho_e(1 - \rho_e)}{n} \times 100\% \qquad (3.24)$$

The linear interpolation method results in the highest performance as in this case the optimization results in the best balance between mass and stiffness. The methods with a high stiffness penalization and no mass penalization (power law and RAMP) show a topology which is very similar to the results of static optimization problems. This shows that the optimization is focused on achieving the highest eigenfrequency by finding the highest stiffness possible while ignoring the effect of the added mass. Methods with a better balance between mass and stiffness penalization show that more mass is located towards the boundary conditions where it has a smaller impact on the eigenfrequency.

This difference is mainly due to the fact that the derivatives of strain and kinetic energy are multiplied with different factors. This multiplication factor comes from the material interpolation method used. Therefore the choice of interpolation method determines the direction of the optimization.

A comparison of the different methods with respect to the requirements for an ideal interpolation method is shown in Table 3.2.

Interpolation method	Result (Hz)	Result with linear (Hz)	M_{nd} (%)
linear	306	306	21.0%
powerlaw	279	286	8.7%
RAMP	277	282	9.0%
polynomial	289	297	15.6%
SINH	267	277	8.5%
Full Sine	293	297	17.6%
Half Cosine	293	296	8.2%
Linear Derivative	283	289	12.8%

Table 3.1: Material interpolation comparison table with optimization results for the benchmark case.

Interpolation method	Penalization	Nonzero derivative	Effect of void material	Local eigenmode frequency
Linear	-	Yes	-	-
Power law	K	No	$\downarrow P_k$	$\downarrow P_k \uparrow \rho_{min}$
RAMP	K	Yes	$\downarrow P_k$	$\downarrow P_k$
Poly	K	Yes	$\downarrow P_k \uparrow r$	$\uparrow r$
SINH	K & M	Yes	$\downarrow P_k$	$\downarrow P_k \downarrow P_m$
Full Sine	K & M	Yes	-	-
Half Cosine	K & M	Yes	$\downarrow P_k$	$\downarrow P_k \downarrow P_m$
Linear Derivative	K & M	Yes	$\downarrow P_k$	$\downarrow P_k \downarrow P_m$
Ideal	equal or -	Yes	$\downarrow P_k$ or -	$\uparrow r$ or -

Table 3.2: Material interpolation comparison table with respect to ideal requirements. The arrows indicate the resulting change when increasing a specific parameter.

3.5. Conclusions

Even with the newly proposed methods, there seems to be no ideal material interpolation method for the optimization of eigenvalue problems. It will remain a trade-off between requirements. However there is the possibility to set up some rules of thumb for choosing the material interpolation method. The choice of the interpolation method determines the focus of the optimization. This could be to optimize for high stiffness, low mass or a good balance between both. A good starting point is summarized below.

- Start of with a linear interpolation method to get an idea of the resulting topology and the maximum performance achievable.
- If the goal is to maximize eigenfrequencies with the best balance of mass and stiffness then apply the full sine method.
- If the goal is to maximize eigenfrequencies with the best stiffness then apply the RAMP method.
- If the effect of void material is too high, reduce ρ_{min} or apply RAMP method.
- If local modes become problematic in the optimization process then apply Pedersen's method 3.16.

All topology optimization problems have local optima and results may differ for other cases. However these conclusions are based on this comparison case and experience with other cases, therefore similar results for other cases are expected.

(a) Linear

(b) Powerlaw $P_k = 3$

(c) RAMP $P_k = 3$

(d) Polynomial $P_k = 3$ $r = 0.5$

(e) SINH $P_k = 3$ $P_m = 3$

(f) Full sine $P_k = 0.5$ $P_m = 0.5$

(g) Half cosine $P_k = 0.5$ $P_m = 0.5$

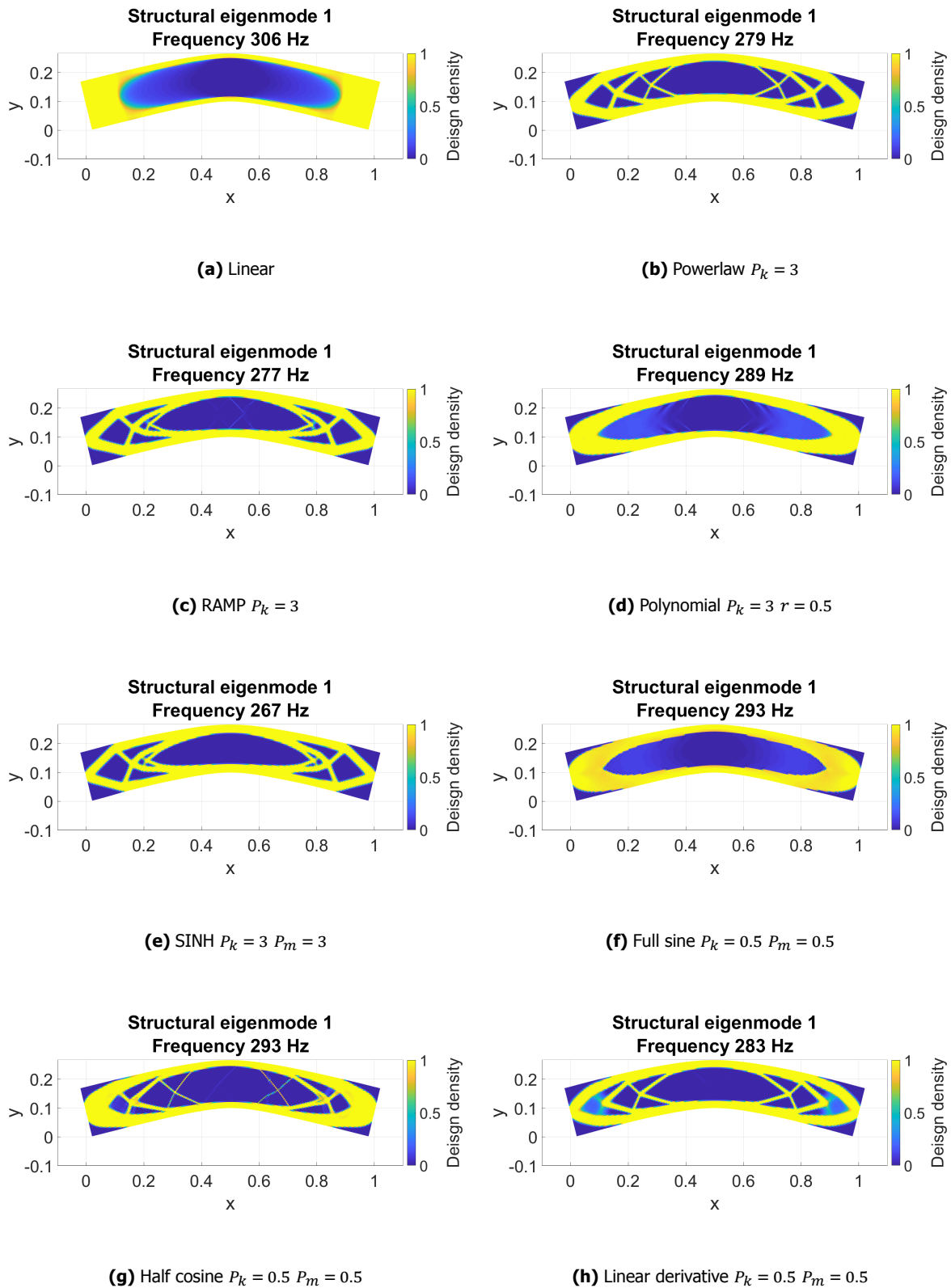**(h)** Linear derivative $P_k = 0.5$ $P_m = 0.5$

Figure 3.8: Material interpolation benchmark case topology results using different methods

4

Mode Tracking and multiplicity

In this chapter the mode tracking and multiplicity problem is described. An algorithm is presented which is able to deal with these problems and is analyzed based on a benchmark case. This benchmark case is then compared with the methods presented in literature. The estimation error of the sensitivity computation is investigated and results are shown.

4.1. Problem definition

When optimizing the eigenfrequencies of a structure, the design is changed at every iteration to improve the eigenfrequency of interest. However by changing the design, all other eigenfrequencies are also affected. Inevitably some eigenmodes will come closer to each other in terms of frequency. For example when optimizing the first bending eigenmode of a structure, the second eigenmode of this structure could be a torsion mode. At some point in the process both eigenmodes have similar eigenfrequencies and the bending eigenfrequency is pushed past the torsion eigenfrequency. In this case the mode shapes have switched order in terms of eigenfrequency, this behavior is commonly referred to as mode switching. When they do not switch but instead go towards the same eigenfrequency, multiplicity of eigenvalues occurs. As a consequence the mode shapes obtained from the eigensolver are no longer unique and form a subspace, any combination of eigenmodes in this subspace is a solution to the eigenvalue problem. This is problematic when tracking the eigenmodes based on their mode shape, also this results in incorrect sensitivity computations.

The eigensolver used to obtain the eigenvalues and mode shapes will provide the eigenvalues in increasing order. In order to compute the correct objective value, constraint values and gradients, the order of mode shapes should remain constant to the optimization algorithm. This ensures that the corresponding eigenvalues can be selected correctly. So an algorithm is required to link the results of the eigensolver to the list of reference mode shapes from the previous or initial iteration.

Methods to track the eigenmodes during the optimization process have already been presented by Kim and Kim [Kim and Kim, 2000] and Gibson [Gibson, 1992]. Both of these methods compare the reference mode shapes to the new mode shapes. The comparison is based on the modal assurance criterion (MAC) shown in Equation 4.1 and the cross-orthogonality check (CORC) shown in Equation 4.2 respectively.

$$MAC(\phi_r, \phi_i) = \frac{|\phi_r^T \phi_i|^2}{(\phi_r^T \phi_r)(\phi_i^T \phi_i)} \tag{4.1}$$

$$CORC(\phi^k, \phi^{k-1}) = \phi^{(k-1)^T} M^k \phi^k \tag{4.2}$$

The MAC relies on comparing the shape of two eigenmodes, the MAC-value represents the projection of the new mode shape on the reference mode shape. This value will be equal to 1 for aligned vectors and 0 for orthogonal vectors. By computing the MAC-value of all eigenmodes with respect to the reference, the order of eigenmodes of interest can be found by searching for the highest MAC values. Tracking n modes requires the computation of n^2 MAC-values.

The cross-orthogonality check uses the mass normalization equation shown in 3.5 as a comparison. Assuming small structural changes, the value of the mass normalization should be close to 1 for the correct pair of eigenmodes and be close to 0 for the wrong pair. This method is however computationally less efficient due to the multiplication with the mass matrix. This method also requires the computation of n^2 pairs of CORC values.

From numerical experiments with the MATLAB eigensolver it has become evident that the problem associated with multiplicity increases as the eigenvalues get closer to one another. For example in the case of a pinned-pinned beam being optimized to have a multiplicity of its first and second bending mode. As the eigenvalues get closer, the first bending mode starts to get a second bending component and the second bending mode gets a first bending mode component. Once they are almost equal in eigenvalue, they seem to have equal components of bending and torsion for both eigenmodes as shown in Figure 4.4a and 4.4b. Both of the mode tracking methods rely on the mode shape information, therefore tracking the eigenmodes at multiplicity becomes problematic.

Besides tracking the eigenmodes at multiplicity, computing the gradient information is also problematic and can be seen from two perspectives. From a mathematical perspective it has been shown that the eigenvalues present in a multiplicity are no longer differentiable using the sensitivity analysis of Section C.1. This is due to the fact that the eigenmodes are no longer unique and any linear combination can be found as a solution to the eigenvalue problem.
However a method by Seyranian et al. [Seyranian et al., 1994] is based on mathematical perturbation, it has been developed to compute the sensitivities of eigenvalues at multiplicity. The method is shown in detail in Appendix E. Following this method, an optimization algorithm for multiple eigenfrequencies was developed by Du and Olhoff [Du and Olhoff, 2005]. This method uses an inner optimization loop to find the optimal step of design density changes. The implementation of this method is complex and requires a lot of computational effort to process. Hence the need for an effective and practical method to deal with multiplicity.

4.2. Theoretical solution

As can be seen from Equation 3.20, the computation of the eigenvalue sensitivities is only dependent on the mode shape and the eigenvalue itself. So as long as the correct mode shapes are supplied, correct sensitivity information should be computed using this equation. The correct mode shapes are present in the subspace of the eigenmodes that are at a multiplicity. Finding these is the key to creating a simple algorithm to deal with multiplicity.
When optimizing a structure, it will change gradually. So if two eigenfrequencies would cross paths, their mode shapes before and after switching should not be that different. Therefore the mode shapes before multiplicity occurs can be used as reference modes to find the correct mode shapes in the subspace.
Finding the correct combination of eigenmodes within the subspace that is similar to the reference eigenmode can achieved by using the MAC-value to compare different combinations. Optimizing this combination to obtain the highest MAC-value is very straight forward and easy to implement. However the algorithm does not know which eigenmodes are forming a subspace, a criteria based on the eigenvalue of the reference eigenmode can be used to select the modes of the subspace. This opens up the possibility to not only deal with the eigenmodes that are at multiplicity, but this works for all eigenmodes that require tracking. When a subspace of unique eigenmodes is selected, the resulting combination from the MAC-value optimization will show only one eigenmode in it.

4.3. Algorithm

A schematic flow chart of the algorithm developed in this thesis is shown in Figure 4.1. The first step in the algorithm is the initialization step, the eigensolver output and reference eigenmodes and eigenvalues are provided to the algorithm.

The first step of finding the correct eigenmode or combination of eigenmodes for the first reference mode is based on the eigenfrequency. Assuming that the eigenvalues change slowly throughout the optimization process, all eigenmodes within a range of $\pm x\%$ of the reference eigenvalue are selected. The value of x is a user input, using a value of 25% seemed to work in most cases. This ensures that the algorithm works properly as long as the eigenvalues do not change by more than 25% after a single iteration.

The second step is analyzing the selected modes, if only one eigenmodes is found within this range then no further action is required and the eigenmode corresponding to the reference mode is found. If no eigenmode is found this means that the eigenvalue has changed too much and the value of x has been set too low. If more than one eigenmode is found, then the MAC-value optimization routine is called.

At this point two or more eigenmodes are found and a multiplicity could be present, so from here they are handled as if they are at multiplicity and the correct combination needs to be found. The MAC-value optimization routine is split in two parts, a least squares (LSQ) fit and the MAC-value optimization. The least squares fit shown in Equation 4.3 is used to provide a starting point for the optimization, without this, the optimization would require many more iterations. The least squares method is cheap in terms of computations required compared to the optimization, however it is does not provide fully correct combinations with optimal MAC-values. That is where the optimization of the MAC-value comes in as shown in Equation 4.4, using the result of the least squares method as a starting point. A second constraint is introduced, by constraining the squared sum of the components the resulting combination satisfies the mass normalization equation.

$$\min_{c_i} \quad ||(\sum_i^n c_i \phi_i - \phi_r||^2 \tag{4.3}$$
$$\text{subject to} \quad -1 < c_i < 1, i = 1, ..., n.$$

$$\max_{c_i} \quad \frac{|\phi_r^T \sum_i^n c_i \phi_i|^2}{(\phi_r^T \phi_r)(\sum_i^n c_i \phi_i^T \sum_i^n c_i \phi_i)} \tag{4.4}$$
$$\text{subject to} \quad \sum_i c_i^2 = 1, i = 1, ..., n.$$
$$-1 < c_i < 1, i = 1, ..., n.$$

The next step is interpreting the optimization results. If no multiplicity of eigenmodes is present in the set of eigenmodes then there should be only one nonzero component c_i. In this case the eigenmode corresponding to the reference mode is found. However if there is a multiplicity of eigenvalues then there are multiple nonzero components, representing the combination with the highest MAC-value with respect to the reference mode. Eigenmodes with a component greater than 0.01 are considered to be part of the multiplicity, this value is selected based on experimenting with the algorithm. The second highest component is checked, when this value is higher than 0.01, the combination of eigenmodes is used.

For every reference mode this algorithm has to be executed once, therefore it is best to track as few eigenmodes as required to reduce the computation time. After every optimization iteration the reference eigenvalues are updated to the new values. Updating the reference eigenmodes is not always required. If the eigenmodes of the first iteration are a good approximation of the eigenmodes of the final iteration, then updating the reference eigenmodes isn't required. If these do change significantly throughout the optimization process, updating the reference eigenmodes is required.

Updating the reference eigenmodes could be problematic if the multiplicity holds on for a longer period of iterations. The obtained combination from the MAC-value optimization might give a good estimation but it will for sure have a small error. When continuously updating the reference eigenmodes this error might build up over several iterations.
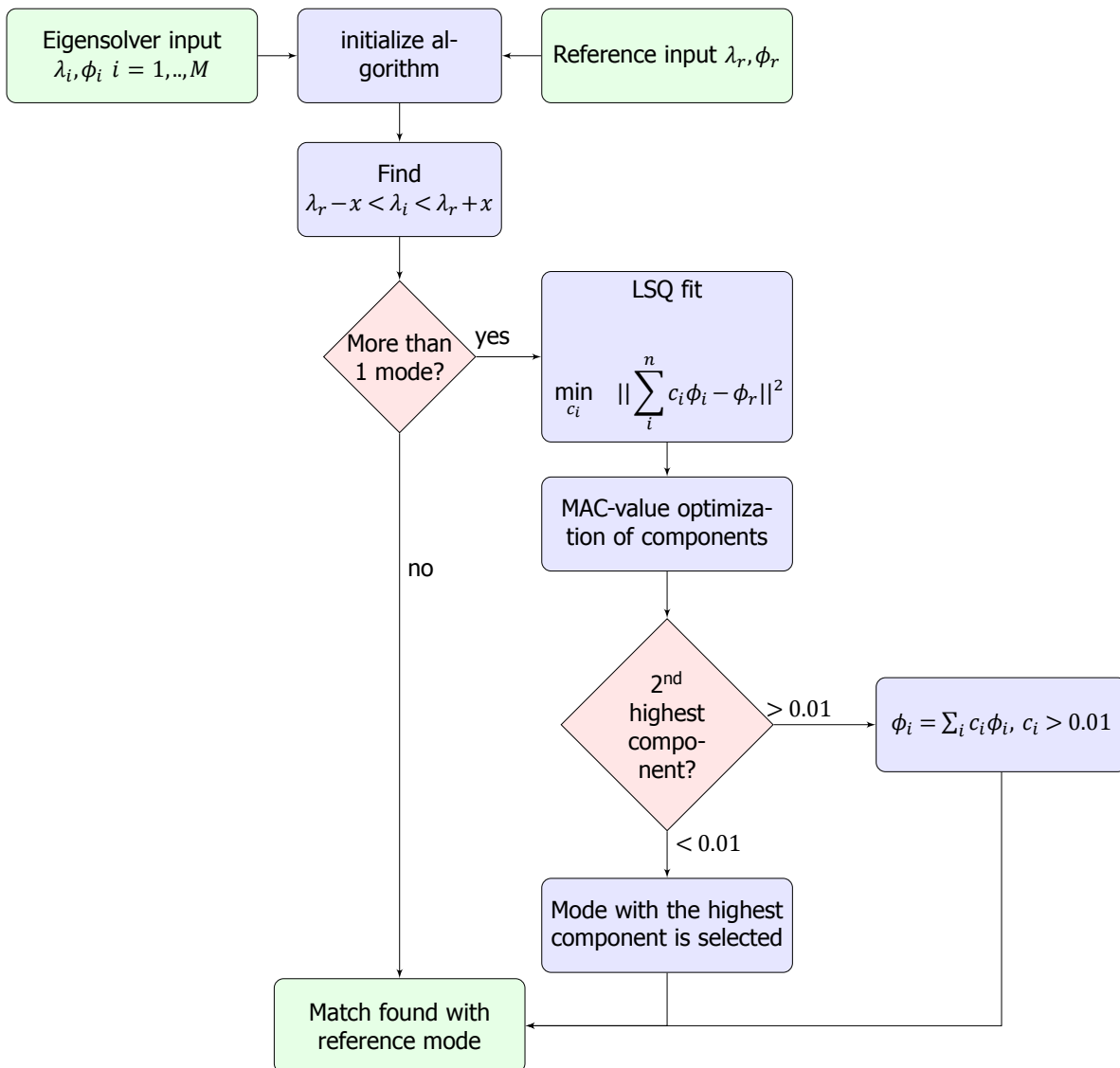
Figure 4.1: Flow chart of mode tracking and multiplicity algorithm

4.4. Comparison of sensitivity computation

The benchmark case used in Section 3.4 has been used to compare the sensitivity computation of the presented algorithm with the analytical method implemented by Du and Olhoff [Du and Olhoff, 2005]. This benchmark case originates from their paper publication. In their paper they demonstrated the topology optimization of the fundamental eigenfrequency, resulting in a multiplicity of the first and second eigenvalue. The resulting topology and eigenfrequency iteration history is shown in Figure 4.2. This shows that their method works correctly and converges to a solution.

The objective is to maximize the fundamental eigenfrequency for a prescribed volume fraction of 50%. The material is isotropic with Young's modulus $E = 10^7$, Poisson's ratio $\nu = 0.3$ and mass density $\rho_{material} = 1$. A power-law material interpolation method with $P_k = 3, P_m = 1$ has been used, however not all details regarding the eigensolver, optimizer and optimization settings are covered in their paper.

(a) Source: [Du and Olhoff, 2005]   **(b)** Source: [Du and Olhoff, 2005]
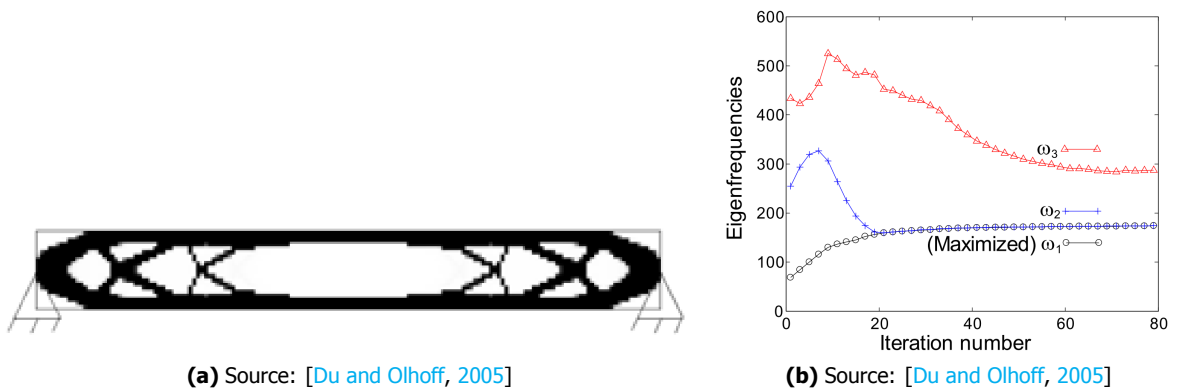
Figure 4.2: Benchmark case results acording to Du and Olhoff [Du and Olhoff, 2005]. (a) Optimized topology for the fundamental eigenfrequency of the benchmark case. (b) Iteration history of the first 3 eigenfrequencies of the structure.

Using the same benchmark case, the proposed algorithm was implemented and the results are shown in Figure 4.3. The final result does not achieve similar performance levels in eigenfrequencies but it does shows a similar iteration history and equal eigenfrequencies of the first and second eigenmode. The reference eigenmodes are not updated iteratively, since the modes of the first iteration closely match those of the final iteration. To demonstrate the effect of the developed multiplicity algorithm, the first and second eigenmodes before and after the algorithm are shown in Figure 4.4.

(a)   **(b)**
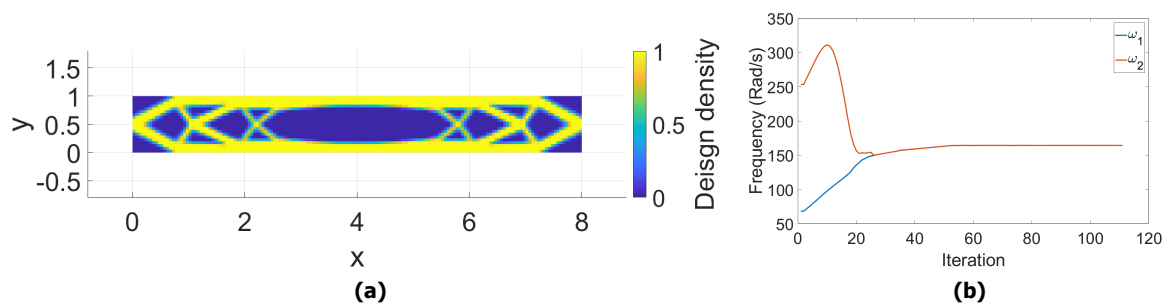
Figure 4.3: Benchmark case results using the method described in this thesis. (a) Optimized topology for the fundamental eigenfrequency of the benchmark case. (b) Iteration history of the first 2 eigenfrequencies of the structure.

(a) Eigensolver output for the first eigenmode. $\omega = 164.3$ Rad/s

(b) Eigensolver output for the second eigenmode. $\omega = 164.4$ Rad/s

(c) Algorithm output for the first eigenmode. $\omega = 164.3$ Rad/s

(d) Algorithm output for the second eigenmode. $\omega = 164.4$ Rad/s
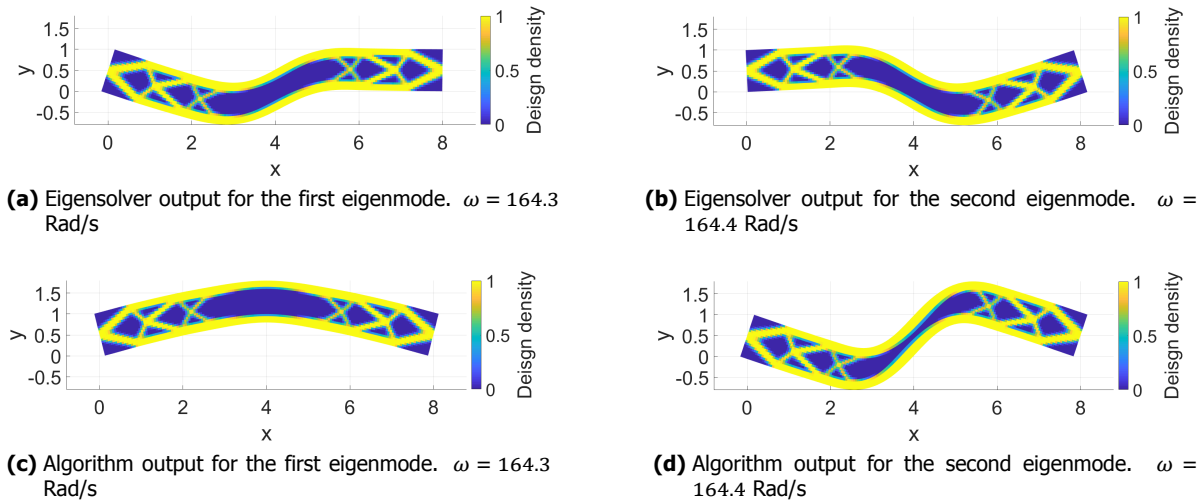
Figure 4.4: Comparison of solver output eigenmodes and algorithm output eigenmodes.

To compare the sensitivity computation of both methods and the FEM results, they are compared using the relative change of the eigenvalues described by Equation 4.5. The $\Delta\lambda_i$ for a single eigenvalue is obtained by multiplying the gradient vector with the step taken by the optimization algorithm. The gradient vector is obtained from Equation 3.20. Using the method described by Seyranian et al. [Seyranian et al., 1994], an eigenvalue delta is obtain for a multiple eigenvalue. The graphs in Figure 4.5 show the relative eigenvalue change for the first 60 iterations of the benchmark case. From these results it is evident that both methods lead to a similar eigenvalue delta. However both methods show a small gap with the results obtained from the FEM analysis, this could be due to non-linearity of the eigenvalue sensitivities.

$$\text{Relative change } (\%) = \frac{\Delta\lambda_i}{\lambda_i} \cdot 100\% \tag{4.5}$$

(a) First eigenmode.
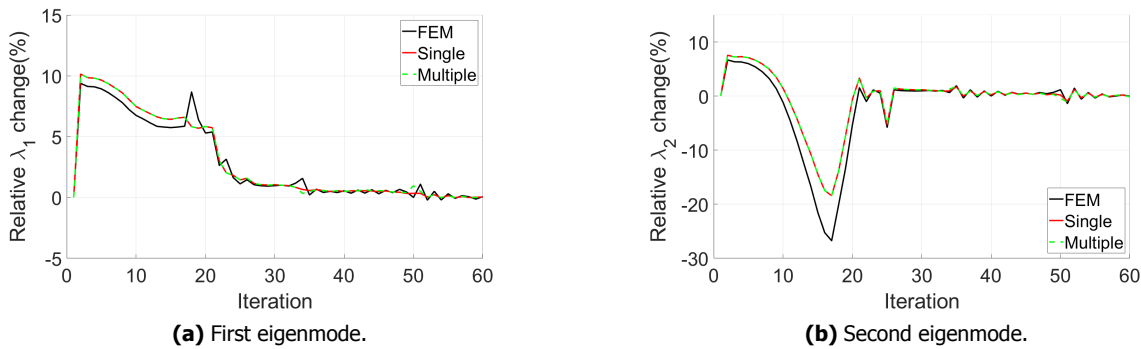
(b) Second eigenmode.

Figure 4.5: Relative eigenvalue changes for the first and second eigenmode of the benchmark case.

4.5. Non-linearity of sensitivity computation

From the iteration history of the relative eigenvalue change shown in Figure 4.5, it becomes clear that there still is a gap between the estimation by using the gradient values and the actual change of the FEM model. Some of the outliers can be related to numerical errors in the FEM analysis, however there is a constant difference between the FEM results and the sensitivity analysis predictions in

the first 20 iterations. This is also the part of the optimization process where the geometry drastically changes, resulting in a small change of the mode shape. The relative change of the MAC-value along with the difference between the FEM $\Delta\lambda_i$ and sensitivity computation delta is shown in Figure 4.6.

The estimated change of the first eigenmode has a relatively constant error throughout the optimization process. The estimation error of the second eigenmode grows substantially throughout the first few iterations. This growing error is caused by the non-linearity in the eigenvalue sensitivity analysis. The sensitivity analysis is based on the mode shape, so if this were to change significantly from one iteration to the next, then the eigenvalue sensitivity analysis has a non-linear component. This effect can be clearly seen in the iteration history of the second eigenmode, the error grows when the MAC-value changes in the optimization process.

This effect can be explained by perturbing the eigenvalue problem as shown by Ki-Ook and Wallerstein [Ki-Ook and Wallerstein, 1988]. The resulting nonlinear perturbation equation is shown in 4.6. When structural changes are assumed to be small $\Delta\phi_i$ is near zero and the same equation (3.20) as in the sensitivity analysis of single eigenvalues is obtained. However if the changes affect the mode shape, $\Delta\phi_i$ becomes nonzero and a difference can be observed.

$$\Delta\lambda_i = \frac{\phi_i^T \left(\Delta K - \lambda_i \Delta M \right) \left(\phi_i + \Delta\phi_i \right)}{\phi_i^T \left(M + \Delta M \right) \left(\phi_i + \Delta\phi_i \right)} \tag{4.6}$$

(a) First eigenmode.
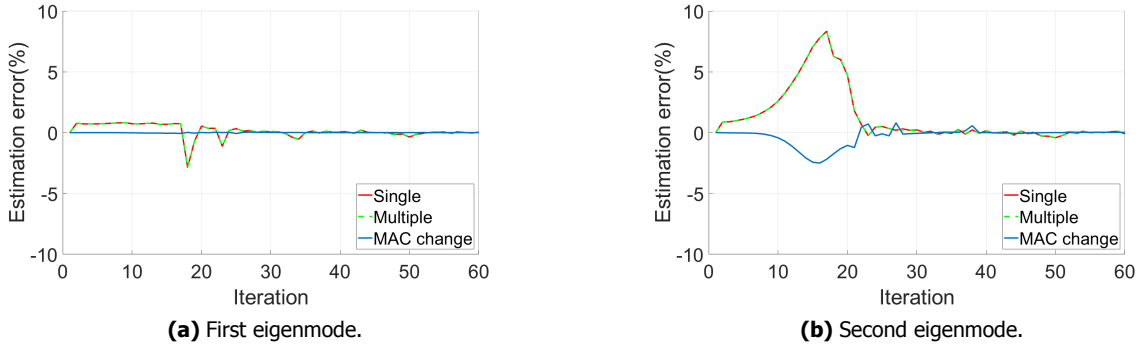
(b) Second eigenmode.

Figure 4.6: Eigenvalue estimation error for the first and second eigenmode of the benchmark case (red: algorithm error, green: analytical method error) and the MAC-value change (blue).

4.6. Conclusions

The presented algorithm shows similar results as the analytical approach presented by Du and Olhoff. The performance numbers are slightly off but this could be caused by differences in the optimization algorithm and finite element mesh. however the resulting topology is similar. The sensitivity information computed with both methods is similar, this shows that the algorithm is able to produce accurate sensitivity information.
Mode tracking is always required when optimizing the eigenfrequencies of a structure. Using the presented algorithm this can be achieved while simultaneously providing a simple solution for the problems associated with multiplicity. Compared with the analytical approach by Du and Olhoff, this algorithm is easy to implement and requires less computational efforts.

5

Convergence to discrete designs

Topology optimization has the design freedom to produce many different designs. However there are several reasons why a discrete design is preferred. In this chapter the challenge of obtaining a discrete design in combination with the constrained eigenvalue optimization is analyzed. A closer look into the MMA optimization process is discussed and existing and new methods are proposed that should aid in obtaining a discrete design. These methods are compared based on a case and conclusions are drawn from the results.

5.1. Problem analysis

The topology optimization method is able to create complex designs with intermediate density areas and relatively small geometrical features. Interpreting these intermediate densities into a physical model is not always possible, from a manufacturer's point of view it is even more complicated. Therefore a final design with only solid or void areas is desired.

Not all of the designs obtained using topology optimization are manufacturable with existing techniques. 3D printing techniques are able to produce many of the designs but are still not able to manufacture everything, let alone in an efficient manner. Therefore the design freedom of the topology optimization method should be somewhat limited such that a manufacturable design is produced. Each manufacturing method has its own limitations so it is impossible to fulfill all requirements. This analysis will be focused on obtaining a discrete design when optimizing the eigenfrequencies of a structure in combination with constraints on its eigenfrequencies.

The final result of the optimization process can show intermediate densities. This can be either due to a zero gradient value of the design density with respect to the objective function or it is in balance with some constraint, for example the volume constraint. Adding material requires removing it elsewhere which would not be optimal. So in order to get a discrete design, either the gradient information needs to be adjusted or the design densities itself. Methods to change either of these are discussed in Section 5.3.

Care should be taken not to limit the optimizer by adding too many constraints or by big variations in the gradients, this could result in sub-optimal designs or slow convergence rates. Several methods have been presented in literature to obtain a discrete design, they have however not been studied in the setting of constrained topology optimization of eigenfrequencies. The added constraints add more difficulty to the equation.

A fully discrete design is nearly impossible to obtain when constraints are present. When using intermediate densities there is a infinite amount of possible designs but when limiting the design variables to either 1 or 0, the number of possible designs is reduced drastically. Increasing the number of design variables by refining the mesh size opens up more design possibilities which could achieve a higher degree of discreteness.

5.2. Analyzing the MMA process

The process of the MMA optimization method as implemented in this research can be summarized by two steps as follows:

- Obtaining a feasible design. If the cost variables of the constraints are scaled appropriately with the objective function, then the optimizer will focus on the constraints first, making sure that the most expensive constraint is satisfied first followed by all other constraints.
- Maximizing the objective function. Once all constraints are satisfied, the optimizer will focus on the objective function while keeping the constraints satisfied. The optimizer uses the gradient information of the active constraints to ensure that the design of the next iteration fulfills the constraints.

Applying techniques to improve the convergence towards a discrete design will often increase the complexity of the optimization problem. This will slow down both of these steps. A smooth problem without constraints can be optimized in an efficient manner with minimal amount of iterations. Applying penalization or filters changes the smoothness of the problem, resulting in a reduced convergence rate. This effect can be reduced by first allowing the optimizer converge on the original smooth problem and at that point apply a method to improve the discreteness of the design. This would be the third step in the MMA process:

- Converging towards a discrete design. After partly converging on the objective function, the focus can be shifted towards optimizing the discreteness.

5.3. Methods to achieve a discrete design

Different methods to improve the optimization results have been presented in literature. In the subsequent sections these will be addressed. Obtaining a discrete design can be achieved in different ways. The working principle of the methods presented here can be divided in two categories:

- Change the design densities. Density based filters are based on changing the design densities and have an impact on the design freedom in term of minimum feature size and transition rate of the boundaries between void and solid.
- Change or add gradient information. The material interpolation methods and objective function based methods change the gradient information of the objective function whereas constraints add gradient information to the optimizer.

5.3.1. Constraint based

A discrete design can be obtained by applying a constraint on the design variables, constraining a large amount of design variables to only be 1 or 0. The added gradient information of this constraint will force intermediate densities towards either solid or void.

SRV constraint

The sum of reciprocal variables (SRV) constraint was introduced by Fuchs et al. [Fuchs et al., 2005] as a method to push the design towards a discrete final result. The SRV value can be computed using Equation 5.1. With a specified volume for the design domain this value can also be computed for a discrete design. Together with the computed discrete value, a constraint can be formed as shown in Equation 5.2. For numerical reasons a relaxation parameter η is added. The downside of this method is the required specific volume. The sensitivity analysis of this constraint is presented in Section D.5.

$$SRV(\rho) = \sum_e^n \frac{1}{\rho_e} \qquad \rho_{min} \leq \rho_e \leq 1 \tag{5.1}$$

$$1 - \frac{SRV(\rho)}{\eta SRV_{discrete}(V)} < 0 \tag{5.2}$$

Smooth discreteness constraint

Inspired by the idea of the SRV constraint, a different constraint is presented here. The idea is to use a constraint to force the design density values towards either solid or void. The sum of the squared shifted design density values is used as the constraint value as shown in Equation 5.3. This value will be equal to 1 when the design is discrete and 0 when only intermediate density elements with 0.5 value are present. The constraint is shown in Equation 5.4, the η parameter is used to define the portion of design variables that should be at a discrete value. The convergence rate towards a discrete design can be controlled by gradually increasing this parameter. The sensitivity analysis of this constraint is presented in Section D.6.

$$SD(\rho) = \frac{\sum_e^n 4(\rho_e - 0.5)^2}{n} \qquad 0 \leq \rho_e \leq 1 \tag{5.3}$$

$$\eta - SD(\rho) < 0 \tag{5.4}$$

Non-smooth discreteness constraint

The downside of the smooth discreteness constraint is the fact that the derivative of the quadratic function goes to zero at 0.5 design density value. A more aggressive approach in the form of a non-smooth discreteness constraint is presented here using a similar equation but employing the absolute value instead of a quadratic value. Equation 5.5 shows the computation of the non-smooth version value. The sensitivity analysis of this constraint is presented in Section D.6.

$$NSD(\rho) = \frac{\sum_e^n 2|\rho_e - 0.5|}{n} \qquad 0 \leq \rho_e \leq 1 \tag{5.5}$$

5.3.2. Filter based

Density filtering is often used to prevent checkerboarding issues, ensure mesh independent results and to provide some control over the feature sizes. These filters, as described in Section 2.7.1, work as a weighted average within a certain radius. The radius has a direct impact on the feature size that is possible in the design domain. This filter radius gives the opportunity to ensure a manufacturable design. Density filtering in itself does not improve the discreteness of a design, it actually blurs all boundary regions between void and solid material due to the averaging nature.

Heaviside filter

In an attempt to create fast transitioning boundaries, the heaviside filter was introduced by Guest et al. [Guest et al., 2004]. After using the basic density filter, the filtered densities are transformed by a smooth heaviside function as shown in Equation 5.6. By increasing the β parameter the function will go towards a step function when it has reached infinity. The effect that this parameter has on the design is visible in the boundary regions. By increasing the β parameter the boundary regions show a faster transition from solid to void producing a clear separation line. This filter will act like a maximum operator on the neighborhood of an element when the value of β is high, resulting in a value of 1 if any of the surrounding elements have a design density larger then the minimum value.

$$\bar{\rho}_e = 1 - e^{-\beta\tilde{\rho}_e} + \tilde{\rho}_e e^{-\beta} \tag{5.6}$$

Modified heaviside filter

A modified version of the heaviside filter presented by Xu and Yuan [Xu and Yuan, 2004] works similarly but has the step function at a design density value of 1 instead of zero. Instead, it works as a minimum operator on the neighborhood, resulting in a value of 1 only if all surrounding elements have a value of 1. Equation 5.7 shows the modified equation.

$$\bar{\rho}_e = e^{-\beta(1-\tilde{\rho}_e)} - (1-\tilde{\rho}_e)e^{-\beta} \tag{5.7}$$

5.3.3. Material interpolation based

The objective of the penalization methods is to improve the discreteness of the final results. Different methods have been presented in Section 3.3, by increasing their penalization factors, the intermediate density elements are less efficient than solid or void elements resulting in a change of design towards a discrete design. The penalization methods change the balance of the kinetic and strain energy derivatives and in that way also effect the sensitivity information of the objective function.
This can be applied from the first iteration or gradually increased during the optimization process as in a continuation method. Applying it from the start can result in a sub-optimal design. The optimization of eigenvalues in most cases is not a convex problem, the final result depends on the starting design. Adding a penalization method with high penalization factors only increases the dependence on the starting design

5.3.4. Objective function based

Instead of applying a constraint to force the design density values to void or solid, a modified objective function could be used. By multiplying the objective function with the smooth or non smooth discreteness value as shown in Equation 5.8, the maximized objective function becomes a balance between a discrete design and a maximized eigenvalue. Multiplication is one method to combine the two objective functions, many more are possible. The power α is used to control the magnitude of the effect. When α is set to zero, it is not taken into account at all, increasing the value past 1 increases the importance of the discreteness objective as its value lies between 0 and 1. In a continuation approach α would be set to a low value and gradually increased to 1 or higher when desired. A sensitivity analysis of this modified objective function is shown in Appendix C.3.

$$f_{obj,c} = (N)SD(\rho)^{\alpha} \cdot f_{obj} \tag{5.8}$$

5.4. Comparison

In order to compare the different methods presented, a benchmark case is used. Figure 5.1 shows the geometry of the problem considered. A square design domain is used with a point mass p in the center, this point mass is equal to 25% of the full solid design domain mass. The area in red surrounding the point mass is set to be solid, representing the area used to connected the point mass to the model.
The objective is to optimize the fundamental eigenfrequency, in this case it will be a multiplicity from the start due to the symmetry of the model. The first three eigenmodes of a full solid design domain are shown in Figure 5.2. The third eigenmode is constrained to have a ratio of 2 with respect to the first eigenmode. Furthermore the design volume is constrained to 50% of the original design volume.

As a base line the problem is optimized using a linear interpolation method without any additional measures to improve the discreteness of the result. The resulting topology is shown in Figure 5.3a, Table 5.1 shows the corresponding first eigenfrequency, ratio of eigenmode 1 and 3, measure of non-discreteness (Equation 3.24) and iterations required. The optimized result of the basic linear interpolation shows quite a large area of intermediate densities.

The results of all other methods are obtained using the same linear interpolation except for the material interpolation methods. Sensitivity filtering is applied to prevent checkerboarding, a small filter radius of 1.5 times the mesh size is used. All methods are applied in a continuation method, gradually increasing their importance after partly converging on the objective function (10 times normal convergence criteria on the step size). The topology results of all methods are shown in Figure 5.3 with the corresponding properties shown in Table 5.1.
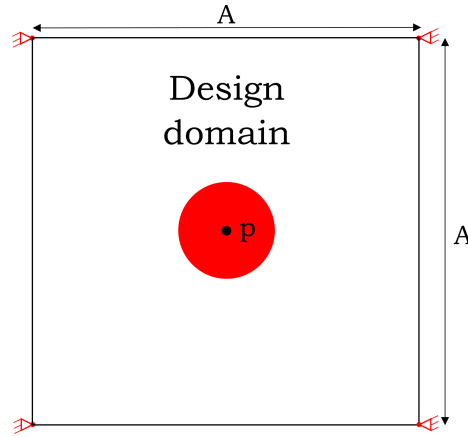
Figure 5.1: Boundary conditions and design domain dimensions for convergence the comparison case.
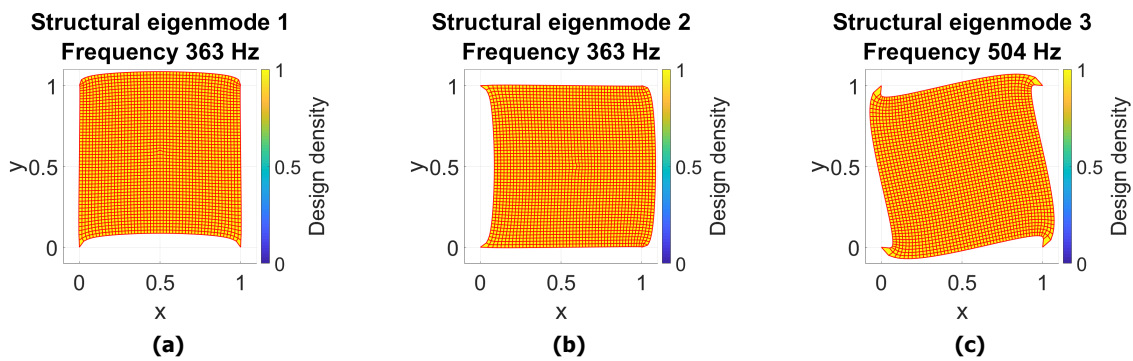
Figure 5.2: First three eigenmodes of the convergence comparison case.

Method	$\omega_1 (Hz)$	ω_3/ω_1	MND (%)	Iterations
Basic linear	353	1.9990	46.2%	208
SRV constraint	336	2.0038	13.1%	310
Smooth bump constraint	339	1.9994	6.3%	407
Non-smooth bump constraint	336	1.9990	18.3%	232
Heaviside filter	352	1.9992	46.0%	232
Modified heaviside filter	353	1.9997	45.6%	376
RAMP interpolation	291 (315)	2.0007 (1.9578)	10.8%	522
Half Cosine interpolation	324 (338)	1.9976 (1.9683)	9.0%	520
Full sine interpolation	340 (349)	1.9990 (1.9849)	24.0%	291
Linear Derivative interpolation	317 (329)	1.9975 (1.9746)	9.7%	530
Smooth discreteness objective	306	1.9966	5.1%	246
Non-smooth discreteness objective	322	1.9973	9.8%	257

Table 5.1: Convergence comparison case results. Results displayed within brackets () are evaluated using a linear interpolation method.

Constraint based methods

The results of the constraint based convergence methods all show an improvement in terms discreteness of the final design. The results are obtained by applying a continuation method on the η parameter

(a) Linear interpolation

(b) SRV constraint

(c) Smooth discreteness constraint

(d) Non-smooth discreteness constraint

(e) Heaviside filter

(f) Modified Heaviside filter

(g) RAMP interpolation

(h) Half cosine interpolation

(i) Linear derivative Interpolation

(j) Full sine interpolation

(k) Smooth discreteness objective

(l) Non-smooth discreteness objective
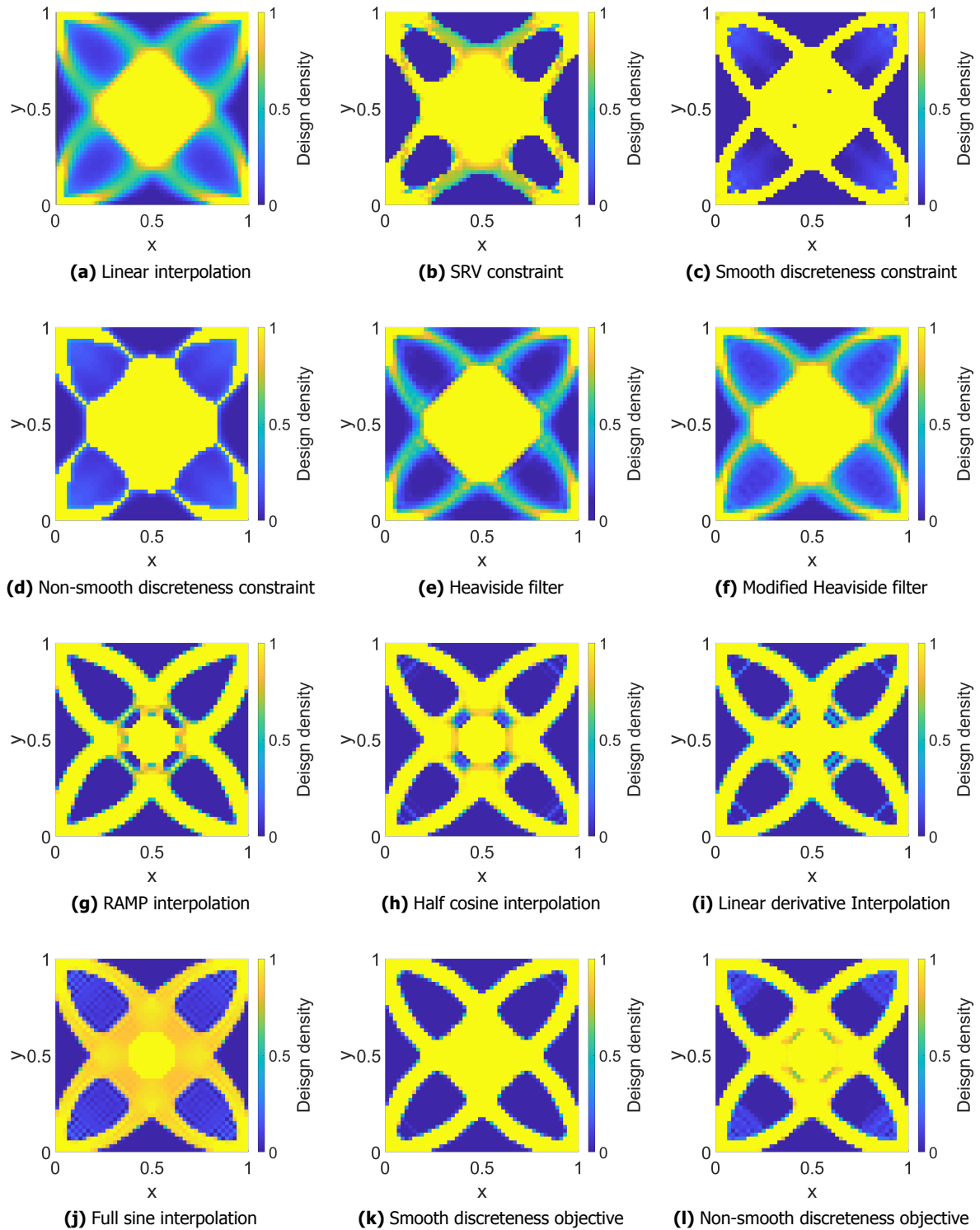
Figure 5.3: Convergence comparison case topology results.

of the constraints. By gradually increasing this parameter the convergence rate of the measure of non-discreteness is controlled. This results in an improvement of the discreteness of the final result without affecting the performance a lot and satisfying the ratio constraint.

The SRV constraint works but does not provide a fully solid result. This could be due to the gradient values of the constraint, these vary significantly in magnitude throughout the design density domain.

The goal of the constraint is to achieve a value for the sum of reciprocal values of the element design densities. The elements with a low design density value have a much larger contribution to the constraint as it uses reciprocal values. That is why the void area's are clearly void but the solid area's are not fully solid. It seems that this result has the highest sum of reciprocal values that satisfies the other constraints.

The smooth discreteness constraint shows a clear solid geometry, the gradient values of this constraint have a much smaller variation throughout the design density domain as it is a quadratic function with a root at 0.5. The benefit of this constraint compared to the SRV constraint is the focus, instead of focusing on the lower density values, it is focused evenly on both sides of the design density domain. It requires a certain amount of elements to be near or completely void or solid depending on the η parameter value. It does however leave small areas of intermediate densities, this has to do with the upper limit set on η which has to be found iteratively. Using a value of 1 for η, meaning all elements should be either solid or void, would not work on this case as it is impossible to reach by the optimizer, resulting in many additional iterations to satisfy all constraints.

The non-smooth version of the discreteness constraint does not perform as well as the smooth version in term of a smooth solid geometry, it creates thin structures and leaves more intermediate densities. It seems that the discontinuous gradient of the function is too aggressive on the intermediate densities below 0.5 to allow them to get to solid. It simply works too good in forcing elements below 0.5 to void and above 0.5 to solid.

Filter based methods

The filter based methods don't actually show much of an improvement to the result, they are in fact almost similar to the unfiltered result. Sensitivity filtering is applied here as well, however similar results were found without sensitivity filtering. The β parameter of the filters is gradually increased up to 100 after reaching partly convergence. Applying a filter doesn't prevent the occurrence of intermediate densities or provides a reason for the optimizer to force a discrete design, it only shifts the design density value at which they occur. A material interpolation function with penalization would be required to force a discrete design. The real Heaviside function can only be obtained by increasing the β parameter to infinity which would change the problem to a discrete optimization problem.

Material interpolation based methods

The use of a material interpolation method with penalization does what it is supposed to do, penalize the intermediate densities. The RAMP, half cosine and linear derivative method all improve the discreteness of the design significantly and converge to a similar geometry. The full sine method does improve it somewhat but still has a high measure of non-discreteness, this due to the fact that the penalization is focused only on the intermediate density region and is less effective close to 0 and 1. The performance of the RAMP method is lower compared to the other 3 methods, this is caused by the focus on maximizing stiffness with the RAMP method. The other methods result in a better performance by penalizing mass and stiffness equally, resulting in a better balance between both. A downside of using penalization is the physical properties, the resulting ratio between the first and third eigenmode is off significantly when the design is evaluated in a linear interpolation model which would represent the physical model.

Objective function based methods

Combining the discreteness function with the objective function provides what seems like the best option for this case. By combining the two the optimizer will find the best balance between a discrete design and the maximized objective function. The value of α is gradually increased from a small value to 1 when the design has partly converged, shifting the focus from maximizing the eigenvalue to obtaining a balance between highest eigenvalue and a discrete design. The lowest value of the measure of non-discrete is obtained using the smooth discreteness function. Another benefit here is that it doesn't necessarily require a parameter study to find the best suited values.

5.5. Conclusions

The general conclusion from comparing different methods is that the optimization problem has to be changed in order to achieve a discrete design as a result. This can be achieved by changing the objective function, applying penalization or adding an additional constraint on the design density variables.

Filtering is useful to control feature sizes but will not aid in converging towards a discrete design. It can however be combined with any of the other methods to control the feature size, in that case the boundary transition rate from void to solid can be improved by applying the heaviside filters.

Applying material interpolation methods with penalization is useful when there are no strict requirements on the ratios between eigenfrequencies. When considering the linear interpolation functions as "physical" reality, the results of the penalized methods have different eigenfrequencies in reality. The magnitude of the penalization has a direct link to this effect. So in the case of strict requirements on the eigenfrequencies, design and reality do not match up. The continuation method required for some cases also add a lot of iterations, every time it is updated the physical properties of the model change. This affects constraints which have to be satisfied in additional iterations.

Constraints are useful when trying to control the convergence rate of the discreteness. However it does not combine with the MMA process very well. Every time the constraint parameter is updated to increase the discreteness, the optimizer has to go through the loop described in Section 5.2, adding additional iterations to the process. A correct maximum value of the discreteness is required for the constraint to achieve an optimal design, when this is set too high the optimization process will be locked in the constraint satisfying step. This value can't be predicted and has to be found iteratively. Since it has to be found iteratively it will always be lower then the maximum possible.

The discreteness objective function provides a robust method of finding the optimized discrete design with maximized eigenvalue. It is easy to implement in any optimization problem and is controlled by a single parameter. Is has the benefit of not affecting the original problem in terms of constraint values or eigenfrequencies, it only affects the objective value and its gradient. This method works very well in combination with the MMA optimization process. Based on the results from this comparison it seems to be most suited for optimization of eigenvalue problems with strict constraints on eigenfrequency ratios. Feature size control can be achieved in some degree using sensitivity filtering, however density filtering remains the better alternative when feature size control is required.

6

Case studies

6.1. Optimization problem description

As introduced in Chapter 1, the design of an AFM cantilever will be used as a case study to demonstrate the practical application of the methods presented in this thesis. Two different cases are defined for the different configurations of the AFM as shown in Figure 1.1. The optimization objectives and constraints for both cases are:

- Fixed tip mode.
  - Maximized fundamental frequency.
  - Eigenfrequency multiplicity of the first bending and torsion mode.
  - Stiffness of 0.9 N/m in bending direction measured at the tip.
  - Additional: eigenfrequency multiplicity of the second bending and torsion mode, at an integer ratio frequency with respect to the first eigenmode.
- Free tip mode.
  - Maximized fundamental frequency.
  - Integer ratio eigenfrequency of the first and second bending mode.
  - Stiffness of 0.9 N/m in bending direction measured at the tip.
  - Stiffness of 25 N/m in longitudinal direction measured at the tip.
  - Additional: integer ratio eigenfrequency of higher bending modes with respect to the first mode.

Figure 6.1 shows two electron microscope images of a cantilever used in practice, many other sizes and shapes are used but this will be used as a reference. The design freedom is limited to a 2D design of the cantilever extruded in the thickness direction in order to manufacture it with existing and affordable methods. This allows for a planar design of the cantilever with no variations in the thickness.
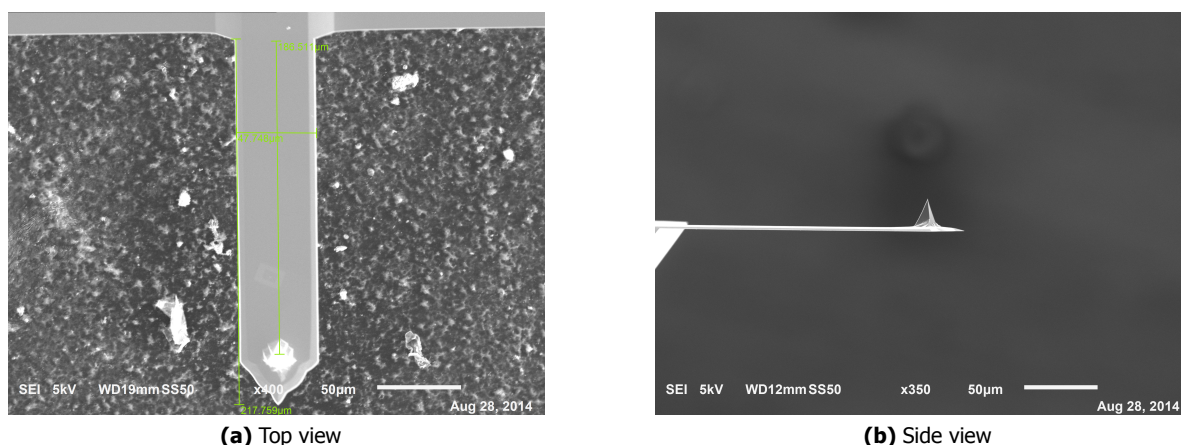

(a) Top view

(b) Side view

Figure 6.1: Electron microscope images of an AFM cantilever.

6.2. AFM cantilever modeling

The boundary conditions as described are shown in Figure 6.2 for both configurations. The tip in the fixed tip configuration is constrained in translation but free to rotate. The free tip configuration is similar to a one sided clamped cantilever. The material properties of Silicon will be used as these tips are commonly made of that material. The outer dimensions of the base cantilever used are 200x50x1.6 μm.

(a) Fixed tip boundary conditions   **(b)** Free tip boundary condition
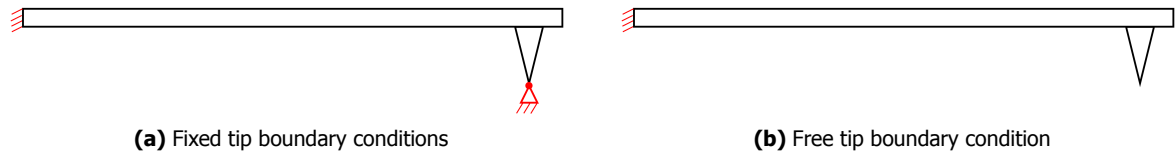
Figure 6.2: AFM tip boundary conditions for the different configurations.

Due to the relatively thin structure of the cantilever, ANSYS shell181 elements with 24 degrees of freedom are used to model the cantilever. A mesh of 80x20 shell elements will be used as shown in Figure 6.3b. The tip of the cantilever is assumed to have a pyramid shape with a base of 10x10 μm and a height of 20 μm in order to compute its center of gravity, mass and inertia. This information is used to add a node at the location of the center of gravity as shown in Figure 6.3a. The mass and inertia of the tip are assigned to this node.

The tip and base of the tip are assumed to be relatively stiff with respect to the cantilever itself, therefore the base area of the tip is set to solid and has its stiffness matrix multiplied by 1000 to represent a much stiffer area. A node is added at the location of the tip in order to compute the stiffness at the location of the tip, no mass is added to this node.

The tip and center of gravity node are connected to the center of the tip base by zero mass beam elements. The modeling of the cantilever and implementation of the finite element code is verified using COMSOL, the results of this comparison are shown in figure 6.4. The results indicate that the modeling and implementation is correct. Table 6.1 shows the tip stiffness, fundamental eigenfrequency and ratios with respect to higher modes for a full solid design domain.

(a) Tip geometry and tip center of gravity (blue dot)   **(b)** Mesh and solid geometry
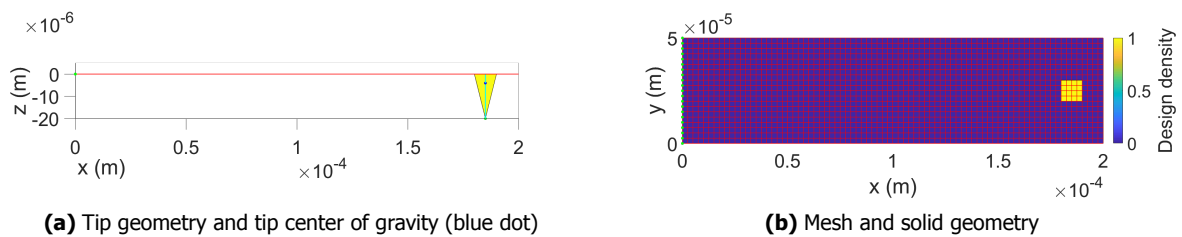
Figure 6.3: AFM cantilever mesh and geometry

Property	Fixed tip	Free tip
ω_1(kHz)	428	51.8
ω_2/ω_1	2.20	6.52
ω_3/ω_1	2.74	8.68
ω_4/ω_1	3.51	18.65
k_z(N/m)	1.41	1.41
k_x(N/m)	40.29	40.29

Table 6.1: Cantilever properties using a full solid design domain.

(a) First bending mode

(b) First bending mode

(c) First torsion mode

(d) First torsion mode

(e) Second bending mode

(f) Second bending mode

(g) Second torsion mode

(h) Second torsion mode

Figure 6.4: Fixed Cantilever modeling comparison of Matlab (right) and COMSOL (left) results

6.3. Optimization methods

Generally it is best to start with the most basic method in optimization and increase the complexity gradually to ensure an optimal design. Therefore the first optimization cycle starts out using the linear interpolation method with all constraints on stiffness and frequency ratios applied. The result of this optimization cycle will give a good indication of what the optimal geometry will look like. In a second cycle sensitivity filtering is applied to remove checkerboarding patterns and small geometrical features. The result should be fairly similar to the unfiltered result. In the final cycle the smooth discreteness objective continuation method is applied to obtain a discrete design. In subsequent cycles additional constraints on the eigenfrequencies are added to improve the design even further.

The stiffness constraints are applied to ensure a solid structure, without these constraints the resulting geometry would have a very weak structure connecting the tip to the boundary conditions and mass located only at the boundary conditions. This would have the best dynamic performance but be very fragile and unusable in practice.

6.4. Fixed tip cantilever with multiplicity

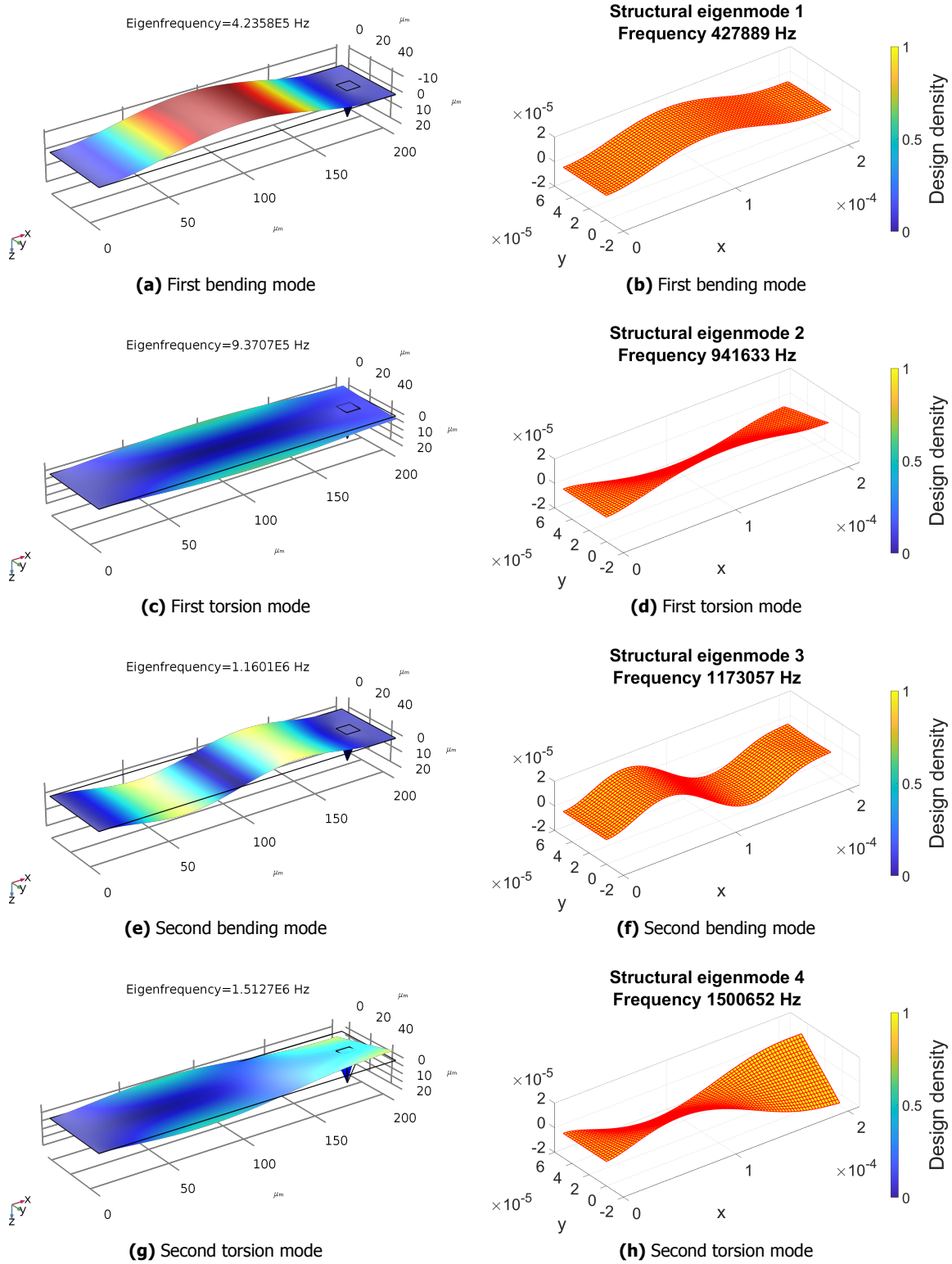As mentioned before the objective of the fixed tip cantilever is to maximize its fundamental eigen-frequency while having a multiplicity of the first bending and torsion mode and a lower limit on the stiffness in the bending direction of the cantilever.
The resulting topology of the basic linear method is shown in Figure 6.5a. The optimized topology for the bending eigenmode would have a minimal amount of mass in the middle section of the cantilever while the optimized torsion geometry would have no mass along the center line and mass at the sides of the cantilever to maximize the torsional stiffness. This result shows a combination of both optimal geometries, balanced such that the eigenfrequencies are equal.
It does show some small geometrical features but those are removed by applying sensitivity filtering as shown in Figure 6.5b. The filtered results also shows a clear geometry but this is improved even more by applying the smooth discreteness objective as shown in Figure 6.5c. The topology obtained using this method can be manufactured and tested in practice. The properties of all results are shown in Table 6.2.

Property	Initial full solid	Basic linear method	Sensitivity filtered	Discrete objective	Double Multiplicity	Double multiplicity at integer ratio	Integer ratio mode 4
ω_1 (kHz)	428	539	540	523	507	493	520
ω_2/ω_1	2.20	1.00	1.00	1.00	1.00	1.00	1.00
ω_3/ω_1	2.74	2.29	2.26	2.16	2.18	2.00	2.17
ω_4/ω_1	3.51	1.98	1.84	2.07	2.18	2.00	2.00
k_z (N/m)	1.41	0.90	0.90	0.90	0.92	0.91	0.90
k_x (N/m)	40.29	21.71	21.33	22.60	23.82	20.53	22.79
V	1	0.71	0.71	0.75	0.78	0.80	0.76
MND	0 %	17.7 %	9.9 %	3.6 %	3.7 %	3.3 %	3.3 %
Iterations		156	185	227	216	226	270

Table 6.2: Fixed tip case study results.

From the results of the discreteness objective method it becomes clear that the third and fourth eigen-mode are also close to each other in terms of frequency. An additional constraint is applied on the third and fourth eigenmode to be at multiplicity as well, resulting in the topology shown in figure 6.5d. It only requires a small adjustment to obtain this multiplicity.

The ratio of the third and fourth eigenmode with respect to the first and second eigenmode is also close to an integer number, therefore an additional constraint is applied on this ratio to be at an integer value of 2. The resulting topology is shown in Figure 6.5e. The optimizer is able to obtain the desired ratios, but in order to reach a lower ratio value there is now a very weak connection between the left and right side of the cantilever. This feature is mainly to reduce the frequency of the second bending mode. In practice this would show high stress concentrations when deformed and therefore have a short lifespan, making this design unusable.

An alternative to this design is obtained by releasing the constraints on the second bending mode, resulting in a structure with multiplicity of the first and second eigenmode and an integer ratio of the fourth eigenmode. The corresponding topology is shown in Figure 6.5f.

(a) Basic linear interpolation mehod.

(b) Linear interpolation with sensitivity filtering.

(c) Linear interpolation with sensitivity filtering and discreteness objective.

(d) Additional: Double multiplicity constraint.

(e) Additional: Double multiplicity at an integer ratio with respect to mode 1.

(f) Additional: Integer ratio of mode 4 with respect to mode 1.
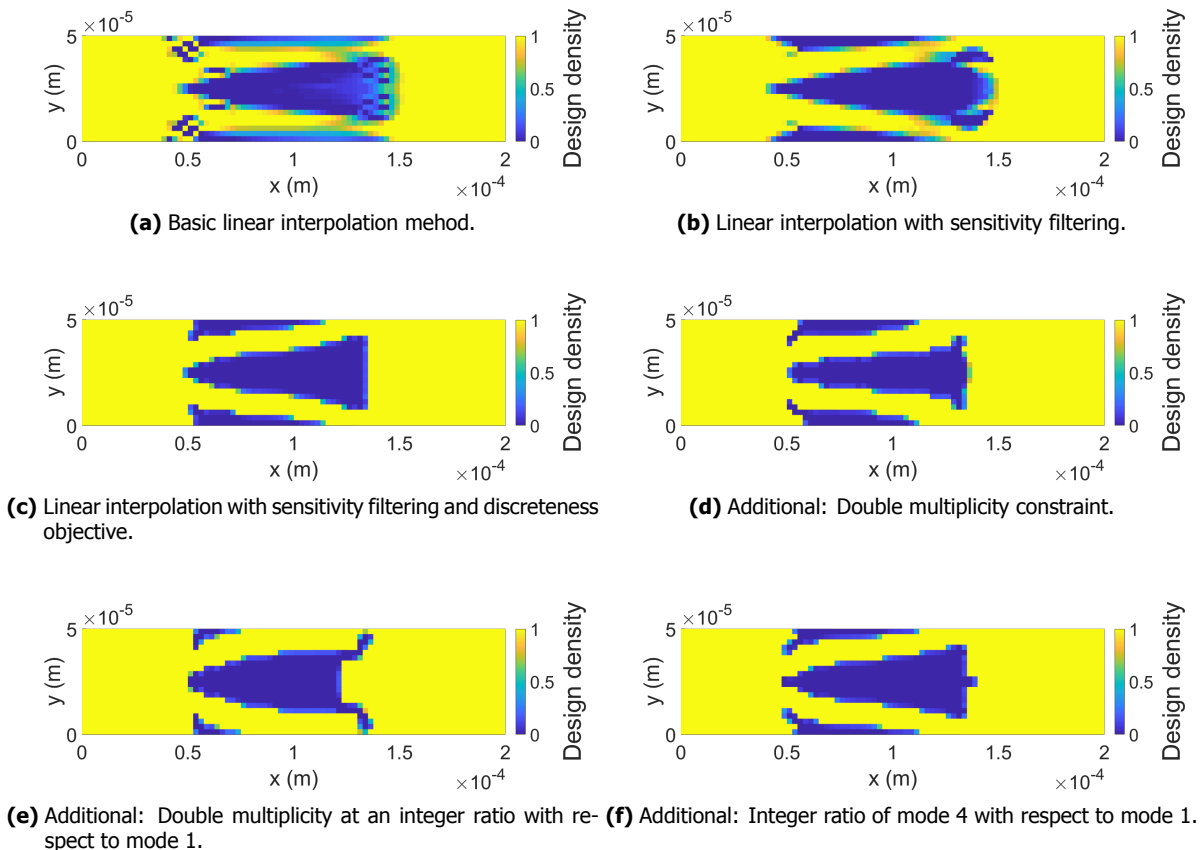
Figure 6.5: Fixed tip case study topology results.

Some general observations can be made based on this case. When constraining two similar eigenmodes such as the first and second bending mode, this often results in local weakening of the structure. In this case the topology shown in Figure 6.5e shows a very weak link between both sides. In the first bending mode this is enough to make it act like one structure but in the second bending mode this area shows a relatively high deformation in the mode shape and acts like a hinge between the left and right side. The stiffness of this hinge determines the eigenfrequency of the second bending mode. The exact geometry and modeling of these hinges is a critical point of the the design, considering manufacturing inaccuracies as well. In order to prevent these critical points, additional stiffness constraints or stress constraints are required.

6.5. Free tip cantilever with integer ratio bending modes

As mentioned before the objective of the free tip cantilever is to maximize the eigenfrequency of the first bending mode while having integer ratios in terms of frequency between the first and higher order bending modes. A lower limit on the stiffness in the longitudinal direction of the cantilever is required in order to obtain a design useful in practice. The first four bending modes are shown in Figure 6.6 for a full solid design.
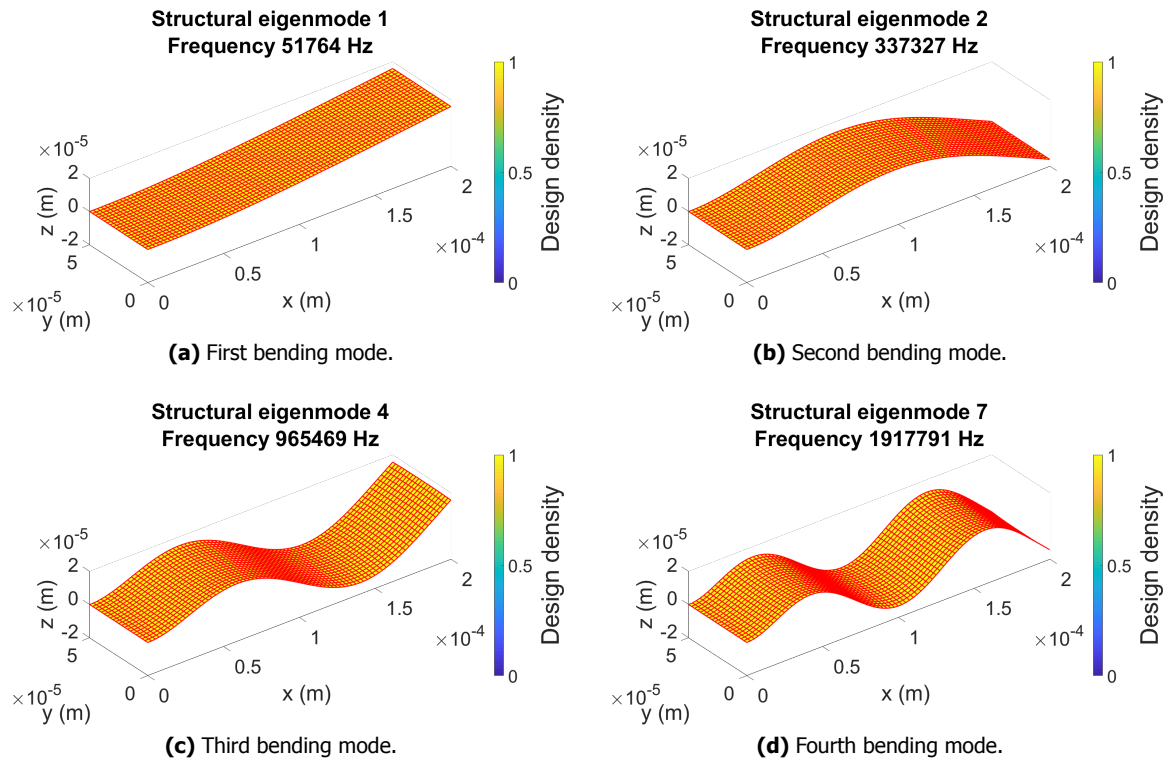

(a) First bending mode.

(b) Second bending mode.

(c) Third bending mode.

(d) Fourth bending mode.

Figure 6.6: Free tip bending modes.

The resulting topology of the basic linear method is shown in Figure 6.7a. The optimized topology for the bending eigenmode of a clamped cantilever would be a very short cantilever. However there is a tip that needs support which requires some stiffness in two direction. These are both opposing each other, increasing the stiffness constraint results in a lower eigenfrequency for the first bending mode, therefore the optimal design is a balance between both.

The result of the basic method already shows a recognizable structure, the sensitivity filtered result shown in Figure 6.7b is not very different as there are no small features or checkerboard patterns to remove.

The final design as shown in Figure 6.7c is obtained using the discreteness objective method. The topology obtained using this method can be manufactured and tested in practice. The properties of all results are shown in Table 6.3. A ratio of 5 between the first and second bending mode eigenfrequency was found to be the lowest feasible, lower values would result in locally weakened structures as seen in the previous case study.

Property	Initial full solid	Basic linear method	Sensitivity filtered	Discrete objective	Integer ratio mode 1,2,4	Integer ratio mode 1,2,4,7
ω_1 (kHz)	51.8	76.2	76.2	75.3	75.3	75.4
ω_2/ω_1	6.52	5.00	5.00	5.00	5.00	5.00
ω_4/ω_1	18.65	12.85	12.85	12.53	13.00	13.00
ω_7/ω_1	37.05	25.12	25.12	25.32	25.59	25.01
k_z (N/m)	1.41	1.33	1.33	1.36	1.36	1.36
k_x (N/m)	40.29	25.00	25.00	25.00	25.68	25.04
V	1	0.73	0.73	0.77	0.76	0.76
MND	0 %	14.3 %	13.1 %	1.0 %	1.0 %	1.8 %
Iterations		329	269	231	244	154

Table 6.3: Free tip case study results.

(a) Basic linear interpolation mehod.

(b) Linear interpolation with sensitivity filtering.

(c) Linear interpolation with sensitivity filtering and discreteness objective.

(d) Additional: Integer ratio of bending mode 2 and 3 with respect to mode 1
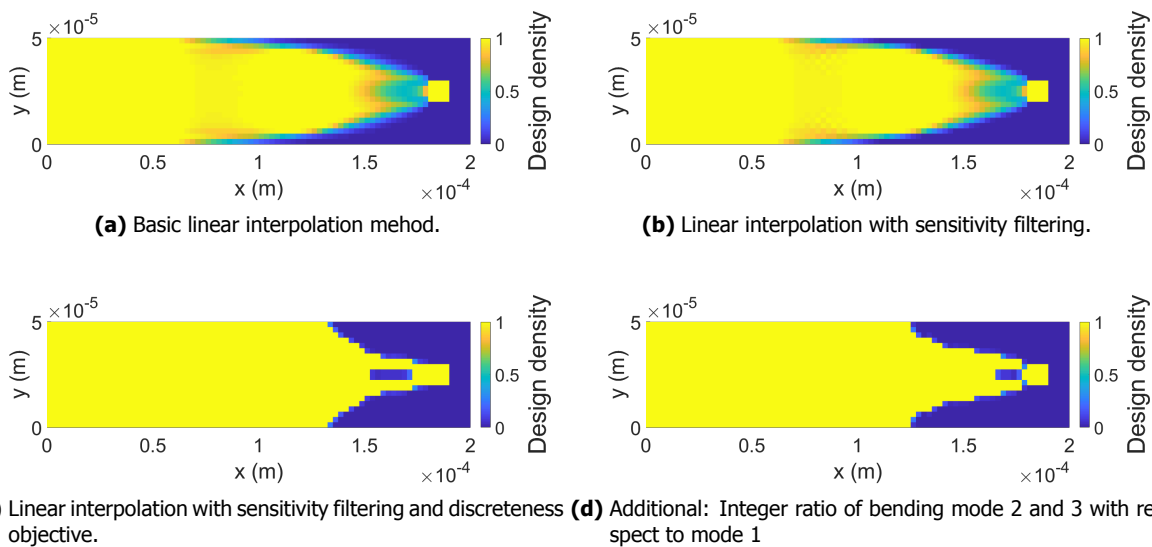
(e) Additional: Integer ratio of bending mode 2, 3 and 4 with respect to mode 1

Figure 6.7: Free tip case study topology results.

From the results of the discreteness objective method it becomes clear that the third bending mode is also close to a integer ratio with respect to the first bending mode. An additional constraint is applied on the ratio of the first and third bending mode to be at a value of 13, resulting in the topology shown in figure 6.7d. It only requires a small adjustment to obtain this ratio, some mesh refinement is required to get a more detailed geometry at the tip. However this would result in small geometrical features which could be problematic to manufacture or show some stress concentrations when deformed.

The ratio of the fourth bending mode with respect to the first bending mode is also close to an integer value, therefore an additional constraint is applied on this ratio to be at an integer value of 25 which is also 5 times the ratio of the second bending mode. The resulting topology is shown in Figure 6.7e. This again shows some intermediate densities around the tip, adjusting the stiffness constraint or changing the mesh size could change this to a manufacturable design.

7

Discussion

Material interpolation methods

The problem of localized vibration modes has been analyzed in depth and a relationship with the material interpolation functions has been found. Local eigenmodes are always present in a structure with clearly separated void and solid volumes. However the effect of the different material interpolation functions on the local eigenmode frequency can be predicted using Equation 3.4. This equation relates the relative change of the local eigenmode frequencies to the material interpolation functions and minimum design density. The relative change depends on the ratio of stiffness over mass at minimum design density, therefore most methods to prevent this effect are based on changing this ratio. Generally the minimum design density should be set as low as numerically allowed to reduce the effect of void material on the structural eigenfrequencies. In order to achieve this without drastically reducing the local eigenmode frequency, a penalization method with equal mass and stiffness penalization at minimum design density is required.

Several eigenmode properties have been analyzed in order to find a suitable local eigenmode indicator. The modal mass appears to be a good indicator, it can be used to distinguish local eigenmodes with a low modal mass from structural eigenmodes with a high modal mass. Equation 3.12 and 3.13 can be used to predict the values around which the modal masses should lie for a structural or local eigenmode. There are however some cases where the modal mass lies in between both, a good interpretation for these eigenmodes is yet to be found.

With the obtained knowledge on the effects of the material interpolation functions on local eigenmodes and the working principles of gradient based optimization, a set of requirements for an ideal material interpolation method for topology optimization of eigenvalues has been set up. Some of these are linked to each other by the relationship for the relative change of local eigenfrequencies, it seems impossible to fulfill all requirements without sacrificing on other requirements. The linear interpolation method shows the least amount of problems but it lacks penalization, this can however be replaced by either a constraint or modified objective function as shown in Chapter 5.

Different existing and new material interpolation methods have been compared using a case often found in literature. The performance of these methods are compared as well as their ability to penalize intermediate densities and produce a discrete design. The method are compared based on the measure of non-discreteness shown in Equation 3.24.
Applying material interpolation functions with penalization has the downside of producing lower performance results compared to linear interpolation, this is due to the fact that elements with a design density lower than 1 have a lower stiffness over mass ratio which is directly linked to the eigenfrequency. The results of the comparison case show that indeed the linear interpolation method is able to produce the best result in terms of maximum fundamental eigenfrequency.
The eigenfrequency of a structure is dependent on the balance between strain and kinetic energy, these are valued equally in the derivatives of the linear interpolation method. All existing and new penalization methods presented here will offset this balance in order to create less efficient elements in terms of performance. This should result in less intermediate densities in the final design but at the same time also lowers the maximum achievable performance. It seems that the material interpolation method is not the best part of the process to try and improve the discreteness of the result without affecting the maximum achievable performance.

The conclusions made here are based on only one comparison case, however the linear interpolation method proved to be a good starting point in numerous tests to get an idea of what the optimal geometry looks like and what performance levels can be expected. When it comes to selecting a material interpolation method to penalize the intermediate densities there seems to be no obvious choice. There are some methods that provide similar results but they all have their own characteristics.

Mode tracking and multiplicity

An algorithm has been presented that is able to deal with mode tracking and multiplicity sensitivity computation simultaneously. It should be implemented as a post processor of the eigensolver results. The order of the eigenmodes is kept equal throughout the optimization process in terms of mode shape, even when eigenfrequencies cross.

When multiplicity occurs, the corresponding eigenmodes form a subspace, any combination of the eigenmodes within this subspace can be used as a solution to the eigenvalue problem. However the computation of the eigenvalue sensitivities is based on the mode shapes, which is why it becomes problematic when a combination of eigenmodes is supplied. The algorithm can be used to find a combination of eigenmodes that represents the reference mode before multiplicity occurred. The normal computation of the sensitivities can be used on this new combination of eigenmodes.

This seems to be a loophole that has not been used in any literature presented. A mathematical solution to the sensitivity computation has been presented by Seyranian et al. [Seyranian et al., 1994] and has been implemented in an optimization algorithm presented by Du and Olhoff [Du and Olhoff, 2005]. Both methods have been compared using a similar case, the topology results of which are similar. This shows that this algorithm can be used as an alternative method of dealing with multiplicity.

The sensitivity computation of the algorithm has been compared to the mathematical approach presented by Seyranian et al. The expected change in eigenvalue at every iteration is compared and found to be similar. This shows that the algorithm numerically works properly. Besides comparing them to one another they have also been compared to the eigenvalue change obtained from the FEM analysis. Both approaches have a similar prediction but also show a gap with the FEM results. Besides a bit of numerical noise there is a substantial gap when the mode shape changes due to the topology changes. This appears to come from the non-linearity in the eigenvalue sensitivities. The degree of non-linearity depends on how fast the mode shape changes, thus a fast changing topology with big step sizes will show a larger estimation error than a slowly changing topology with a more conservative step size. In practice this does not prove to be problematic.

The algorithm should work for all cases if the reference modes of the initial iteration are a good indication of the eigenmodes throughout the optimization. If this is not the case and the geometry drastically changes, updating the reference is required. This is a downside of the method as it is hard to predict whether the initial eigenmodes are a good estimation. More research is required to find a smart updating strategy.

In some cases the initial iteration can already exhibit multiplicity, this commonly occurs in symmetric structures. The eigenmodes that are obtained in this case are probably dependent on the eigensolver and might or might not be the expected eigenmodes. The comparison case shown in Section 5.4 uses a symmetrical structure which has a multiplicity at the initial iteration, different eigenmodes are found depending on the value of the initial design densities. More research is required to develop a strategy to deal with multiplicity of eigenmodes in the initial iteration.

In some cases a strange behavior occurred when two eigenmodes are at multiplicity for a long period of iterations. Numerically the eigenvalues are never equal, so assigning the eigenvalue to each mode is based on the highest components obtained from the MAC-value optimization. But in some cases these kept switching at every iteration, resulting in a somewhat oscillatory behavior of the eigenvalues which is not ideal when using the MMA optimization process. This might be caused by the eigensolver or be case dependent, more research in this behavior is required.

Discrete design convergence

Different existing and new approaches to improve the discreteness of the topology optimization results have been compared. The combination of eigenvalue optimization, the MMA algorithm and specific constraints on eigenfrequencies has not been investigated before. Material interpolation methods with penalization are commonly used to improve the discreteness of the results presented in literature. However as shown here, that does not work very well in combination with the constraints and the desire for maximum performance. The topology that results from material interpolation methods with penalization have different ratios when evaluated using the linear interpolation method. When considering the linear interpolation method as physical reality, this error is undesired.

The results of the comparison case show that the discreteness constraints and objective functions can be used as an alternative to improve the discreteness of the results. The discreteness constraint does however require a parameter study to find the maximum achievable degree of discreteness whereas the discreteness objective function does not. For this reason the discreteness objective function seemed to be the best option of all methods presented.

It has only been tested in combination with topology optimization of eigenvalues but this method might work for all topology optimization routines as it does not affect the original problem. Further research into the application of the method is required but it has the potential to be a substitute of the penalization method using material interpolation functions.

The discreteness objective function has only been tested in combination with sensitivity filtering, combining the method with density filtering seems counter intuitive as that again blurs the boundaries and prevents a fully discrete result. Applying the Heaviside filters might result in faster transitions of the boundaries but this has to be investigated. Sensitivity filtering does provide some control over the feature size but not to the extent of density filtering.
Sensitivity filtering is often considered a heuristic and inconsistent method as the filtered sensitivities are not directly related to the objective function. However in a paper by Sigmund [Sigmund and Maute, 2012] the sensitivity filtering equation is derived from continuum mechanics and nonlocal elasticity concepts. Whether this is also true for eigenvalue problems is yet to be found.

Discrete design convergence

Framework for topology optimization of eigenvalues

Based on the obtained results, a general framework for the constrained optimization of eigenfrequencies is formed and used to optimize the cases introduced in Chapter 1.

- Optimization algorithm

  - The MMA optimization algorithm works well on gradient-based optimization problems and often converges to high-quality optimized designs by gradually changing the structure. Experimenting with different problems, it seems that the optimization of eigenvalues requires a smoother process than static problems due to the required mode tracking. The MMA algorithm is well suited for a smooth optimization process.

- Objective function

  - Depending on the optimization objective, it can be based on a single eigenvalue or mean eigenvalue in case of multiple eigenvalue objectives.
  - To improve the discreteness of the results the objective function can be modified using the smooth discreteness objective function which is very straight forward to implement.

- Constraint possibilities verified to be feasible in this thesis, more options are available in literature.

  - Ratio of eigenvalues.
  - Stiffness at a specific location.
  - Static compliance of a user defined load case.
  - Upper and lower volume limit.

- Material interpolation method

  - As the penalization of intermediate densities is moved towards the objective function, there is no need to use a material interpolation method other than the linear method. At the same time this method is also able to achieve the highest performance in terms of maximized eigenfrequencies. Generally local eigenmodes are not a problem when using the linear method, but those that are present can be detected using the modal mass values.

- Filtering

  - Sensitivity filter has to be applied to prevent checkerboarding patterns, ensure mesh independent results and enables some control on the minimum feature size.

- Mode tracking & Multiplicity

  - The algorithm presented in this thesis can be used to track the order of eigenmodes and simultaneously provides a method to deal with the occurrence of multiplicity of eigenvalues throughout the optimization process. Sensitivity information is computed correctly using the algorithm.

The AFM tip case studies have been optimized using the above setup, the results of which seem feasible to manufacture and test in practice. A more detailed design can be found using smaller mesh sizes and a verification step should be done using a commercial program such as COMSOL to check the eigenfrequencies. But this results shows the feasibility and robustness of the presented setup, minimal optimization cycles are required to obtain the presented results. The only difficulty lies in defining the constraint values for the ratios, these have to be found iteratively. But as shown, these can be easily found by gradually increasing the complexity of the problem, starting out with only one constraint on the frequency ratio.

8

Conclusions and recommendations

8.1. Conclusions

The main goal of this has been to develop a robust framework for the topology optimization of constrained eigenfrequencies. The most important aspects have been analysed and the results have been combined into a framework. The main contributions of this thesis, connecting directly to the goals defined in Chapter 1, are:

- A clear view on the effect of different material interpolation methods on the topology optimization process.

- A simple yet effective method of tracking the eigenmodes during the optimization process combined with a solution for the multiplicity eigenvalue sensitivity computation.

- A new method of obtaining a discrete design without applying penalization or modification of the eigenvalue problem in the form of a smooth discreteness objective function.

Using the presented framework, the case studies of the AFM cantilever have been optimized. Feasible topology results have been found from a manufacturers point of view. The fixed tip cantilever has been optimized with the desired multiplicity of the first bending and torsion mode, additional ratios of eigenfrequencies have been constrained resulting in feasible and infeasible topologies. The free tip cantilever has been optimized with the desired integer ratios of bending modes up to the fourth bending mode. The results of which seem feasible from a manufacturers point of view. From this we can conclude that the presented framework has proven to provide feasible results, in the cases on infeasible results this is due to infeasible constraints.

The common problem of localized eigenmodes has been analysed, a relationship with the material interpolation functions has been found. Several mode shape properties have been analysed to find a sorting criteria for local and structural eigenmodes. From these, the modal mass value proved to be a solid sorting criteria.
With regards to selecting a material interpolation method to use in the optimization process, a set of requirements for an ideal material interpolation method has been set up. However none of the existing and new method presented here can fulfill all requirements. When leaving out the penalization requirement, the linear interpolation method seems the best option that does not show the common problems. It provides the best balance between mass and stiffness resulting in the highest performance. Another reason to use this method is the translation from model to reality. When accurate eigenvalue ratios are desired, material interpolation methods with penalization should be avoided. Selecting one material interpolation method to be used in all topology optimization problems involving eigenvalues seems impossible. However based on the results of this research, the best option seems to be to move the penalization of the intermediate density elsewhere and use the linear interpolation method.

A simple solution to deal with mode tracking and multiplicity simultaneously has been presented. A new algorithm has been developed that is able to deal with mode tracking and simultaneously provides a practical method to deal with the computation of eigenvalue sensitivities when multiplicity occurs. The sensitivities have been compared to an analytical approach and found to be similar. There is still room for improvement in the algorithm but based on the results, we can conclude that the algorithm is a solid addition to the topology optimization framework, it can be implemented in all eigenvalue optimization problems.

Several different existing and new methods of improving the discreteness of the final design have been analysed. The smooth discreteness objective function modification performed best. It can be used a substitute of material interpolation penalization to produce discrete designs. Its straight forward ease of implementation might make this an attractive method for other topology optimization applications as well. Based on the results and experiments with this new method, we can conclude that this new method is a solid replacement of the material interpolation penalization method.

8.2. Recommendations

The proposed framework for the constrained topology optimization of eigenvalues shows promising results based on the presented case studies. The following list proposes some points on interest in the current framework that should be developed further.

- The developed framework has only been tested on the case studies presented in this thesis, setting it up proved to be fairly easy but this might not be the case for other problems. Testing it on more cases should confirm the robustness to deal with all constrained eigenvalue optimization problems. However new points of improvements will always show up.

- The discreteness objective function should be tested on more cases to show that it is a feasible replacement for material interpolation methods with penalization. It should also be tested on different topology optimization problems that involve eigenvalues such as dynamic compliance, frequency response and buckling. The static compliance problems might also benefit from this approach. In theory it should work as it is independent of the underlying problems and is based only on the design density values.

- The presented algorithm for mode tracking and solving the multiplicity eigenvalue sensitivity problem can be improved on several points. Its speed can be improved by using a reduced amount of nodal data for the mode shapes. A smart updating strategy of the reference modes should be developed for cases where the initial eigenmodes are not a good representation of the eigenmodes throughout the optimization. Similarly, the algorithm should be tested more extensively on cases with multiplicity of eigenmodes in the initial design. Finding a set of reference modes is in this case up to the user.

8.3. Future research

Looking at the state of the presented framework, there are always possibilities to add in improvements. The following list proposes some new ideas for improvements.
- Currently sensitivity filtering is applied to prevent checkerboard patterns and allow some control on the feature sizes. This control is only limited and does not prevent small features from showing up. Density filtering provides more control on the feature size and keeps sensitivities consistent but this is avoided due to the blurring of the boundaries. A new method could be developed which enables more control on the feature size without directly affecting the design densities. This could be in the form of a modified version of the sensitivity filter, a constraint based method or a modified objective function method such as the discreteness objective.

- In the AFM cantilever case study stiffness constraints are applied to prevent fragile designs, these designs would have a higher eigenfrequency but would be too fragile to use in the machine. The stiffness constraint is satisfied, but in some cases the structure is locally weakened to get the desired eigenfrequencies. The locally weakened areas also show a relatively large deformation in some mode shapes. Although this does satisfy the stiffness constraint, it is not feasible in practice due to the stresses that would occur in this locally weakened area. The maximum stress or strain could be limited by a constraint, either on the static load case or on the relative values in the mode shapes.

- Within the presented framework there is no control on the mode shapes themselves, this could be useful in certain cases to eliminate specific vibrations in certain directions for some eigenmodes or a fully symmetric mode shape for a non-symmetric boundary condition. It has been demonstrated to some degree by Tsai and Cheng [Tsai and Cheng, 2013], however attempts to recreate this work have not been successful yet. Even though the direct practical applications might be unclear, these will for sure arise when this functionality is added to the presented framework.

References

Andreassen, E., A. Clausen, M. Schevenels, B. S. Lazarov, and O. Sigmund
  2011. Efficient topology optimization in matlab using 88 lines of code. *Structural and Multidisciplinary Optimization*, 43(1):1–16.

BendsØe, M. P. and O. Sigmund
  2003. *Topology optimization : theory, methods and applications*. Berlin ; Heidelberg ; New York ; Barcelona ; Hong Kong ; London ; Milan ; Paris ; Tokyo: Springer.

Bendsøe, M. P.
  1989. Optimal shape design as a material distribution problem. *Structural optimization*, 1(4):193–202.

Bendsøe, M. P. and N. Kikuchi
  1988. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197–224.

Binnig, G., C. F. Quate, and C. Gerber
  1986. Atomic force microscope. *Physical Review Letters*, 56(9):930–933.

Bongwon, J., P. Chris, D. Sajith, K. Hohyun, L. Joohyung, K. Jungkyu, K. Seok, D. M. McFarland, A. B. Lawrence, F. V. Alexander, and C. Hanna
  2016. Utilizing intentional internal resonance to achieve multi-harmonic atomic force microscopy. *Nanotechnology*, 27(12):125501.

Bruns, T.
  2005. A reevaluation of the simp method with filtering and an alternative formulation for solid–void topology optimization. *Structural and Multidisciplinary Optimization*, 30(6):428–436.

Bruns, T. E. and D. A. Tortorelli
  2001. Topology optimization of non-linear elastic structures and compliant mechanisms. *Computer Methods in Applied Mechanics and Engineering*, 190(26):3443–3459.

CCEM
  2018. Atomic force microscope. *https://ccem.mcmaster.ca/atomic-force-microscope/*.

Courant, R.; Hilbert, D.
  1953. *Methods of mathematical physics*, volume 1. New York; London; Sydney: Interscience publishers.

Du, J. and N. Olhoff
  2005. Topology optimization of continuum structures with respect to simple and multiple eigenfrequencies. *6th world congresses of structural and multidisciplinary optimization, Rio de Janeiro, Brazil*.

Du, J. and N. Olhoff
  2007. Topological design of freely vibrating continuum structures for maximum values of simple and multiple eigenfrequencies and frequency gaps. *Structural and Multidisciplinary Optimization*, 34(2):91–110.

Fuchs, M., S. Jiny, and N. Peleg
  2005. The srv constraint for 0/1 topological design. *Structural and Multidisciplinary Optimization*, 30(4):320–326.

Gibson, W. C.
  1992. Astros-id: Software for system identification using mathematical programming.

Guest, J. K., J. H. Prévost, and T. Belytschko
  2004. Achieving minimum length scale in topology optimization using nodal design variables and projection functions. *International Journal for Numerical Methods in Engineering*, 61(2):238–254.

Kato, J., K. Ishii, K. Terada, and T. Kyoya
  2014. Layout design of rockbolts for natural ground reinforcement. *International Journal for Numerical and Analytical Methods in Geomechanics*, 38(3):236–255.

Ki-Ook, K. and D. V. Wallerstein
  1988. Modal design sensitivities for multiple eigenvalues. *Computers & Structures*, 29(5):755–762.

Kim, T. S. and Y. Y. Kim
  2000. Mac-based mode-tracking in structural topology optimization. *Computers & Structures*, 74(3):375–383.

Lancaster, P.; Wittrick, W.
  1964. On eigenvalues of matrices dependent on a parameter. *Numer. Math.*, 6(1):377–387.

Ma, Z.-D., H.-C. Cheng, and N. Kikuchi
  1994. Structural design for obtaining desired eigenfrequencies by using the topology and shape optimization method. *Computing Systems in Engineering*, 5(1):77–89.

Ma, Z.-D., N. Kikuchi, and H.-C. Cheng
  1995. Topological design for vibrating structures. *Computer Methods in Applied Mechanics and Engineering*, 121(1):259–280.

Ma, Z. D., N. Kikuchi, and I. Hagiwara
  1993. Structural topology and shape optimization for a frequency response problem. *Computational Mechanics*, 13(3):157–174.

Osher, S. and J. A. Sethian
  1988. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79(1):12–49.

Pedersen, N.
  2000. Maximization of eigenvalues using topology optimization. *Structural and Multidisciplinary Optimization*, 20(1):2–11.

Rozvany, G. I. N., M. Zhou, and T. Birker
  1992. Generalized shape optimization without homogenization. *Structural optimization*, 4(3):250–252.

Seyranian, A. P., E. Lund, and N. Olhoff
  1994. Multiple eigenvalues in structural optimization problems. *Structural optimization*, 8(4):207–227.

Sigmund, O.
  1994. *Design of Material Structures using Topology Optimization*. Thesis.

Sigmund, O.
  2001. A 99 line topology optimization code written in matlab. *Structural and Multidisciplinary Optimization*, 21(2):120–127.

Sigmund, O.
  2007. Morphology-based black and white filters for topology optimization. *Structural and Multidisciplinary Optimization*, 33(4):401–424.

Sigmund, O. and K. Maute
  2012. Sensitivity filtering from a continuum mechanics perspective. *Structural and Multidisciplinary Optimization*, 46(4):471–475.

Sigmund, O. and J. Petersson
 1998. Numerical instabilities in topology optimization: A survey on procedures dealing with checker-boards, mesh-dependencies and local minima. *Structural optimization*, 16(1):68–75.

Stolpe, M. and K. Svanberg
 2001. An alternative interpolation scheme for minimum compliance topology optimization. *Structural and Multidisciplinary Optimization*, 22(2):116–124.

Svanberg, K.
 1987. The method of moving asymptotes—a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24(2):359–373.

Taylor, J. E. and M. P. Bendsøe
 1984. An interpretation for min-max structural design problems including a method for relaxing constraints. *International Journal of Solids and Structures*, 20(4):301–314.

Tcherniak, D.
 2002. Topology optimization of resonating structures using simp method. *International Journal for Numerical Methods in Engineering*, 54(11):1605–1622.

Tortorelli, D. A. and P. Michaleris
 1994. Design sensitivity analysis: Overview and review. *Inverse Problems in Engineering*, 1(1):71–105.

Tsai, T. D. and C. C. Cheng
 2013. Structural design for desired eigenfrequencies and mode shapes using topology optimization. *Structural and Multidisciplinary Optimization*, 47(5):673–686.

Wang, M. Y., X. Wang, and D. Guo
 2003. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1):227–246.

Xia, Q., T. Zhou, M. Y. Wang, and T. Shi
 2014. Shape and topology optimization for tailoring the ratio between two flexural eigenfrequencies of atomic force microscopy cantilever probe. *Frontiers of Mechanical Engineering*, 9(1):50–57.

Xie, Y. M. and G. P. Steven
 1993. A simple evolutionary procedure for structural optimization. *Computers & Structures*, 49(5):885–896.

Xu, G. and X. G. Yuan
 2004. A new density☐stiffness interpolation scheme for topology optimization of continuum structures. *Engineering Computations*, 21(1):9–22.

Yang, X. Y., Y. M. Xei, G. P. Steven, and O. M. Querin
 1999. Bidirectional evolutionary method for stiffness optimization. *AIAA Journal*, 37(11):1483–1488.

Zhu, J. H., P. Beckers, and W. H. Zhang
 2010. On the multi-component layout design with inertial force. *Journal of Computational and Applied Mathematics*, 234(7):2222–2230.

A

Numerical implementation

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

A.1. Finite elements

The finite element model used in the topology optimization process is generated using MATLAB and ANSYS. MATLAB is used assemble the global mass and stiffness matrix using the element matrices generated by ANSYS. The ANSYS application used for this is ANSYS 16.0 APDL. An input text file is written by MATLAB and subsequently ANSYS is called with the input file, the mass and stiffness matrices are then exported by ANSYS. The matrices are exported into a Harwell Boeing sparse matrix format. Using a MATLAB script written by John Burkardt, the matrices are imported into MATLAB. An example of the text file for a PLANE182 (8 DOF quad element) is shown below.

```
1: /CLEAR,START
2: !Element Parameters:
3: E=10000000
4: nu=0.3
5: Dens=1
6: G=3846153.84615385
7: dx=0.02
8: dy=0.02
9: dz=0.00
10: /Prep7
11: ET,1,PLANE182,,,0,,,
12: !material propperties
13: MP,EX,1,E
14: MP,PRXY,1,nu
15: MP,DENS,1,Dens
16: MP,GXY,1,G
17:
18: ! define volume
19: BLC4, 0, 0, dx, dy, dz
20:
21: ! Meshing of volumes
22: TYPE,1  !element type1
23: DESIZE, 1, 1, 1, 15, 28, 1, 1,,
24: AMESH,ALL
25:
26: ! Start modal analysis
27: /SOLU
28: antype,modal
29: modopt,lanb,1,,,,1 ! normalized to unity
30: solve
31: /output
32:
33: FINISH
34: /AUX2
35: file,file,full,
36: ! export mass and stiffness matrices
37: hbmat,HBMstiff,txt,,ascii,stiff,yes,yes
38: hbmat,HBMmass,txt,,ascii,mass,yes,yes
39: FINISH
```

Line 1-11 is written by MATLAB, containing the material properties, dimensions, and element type. The rest of the text is similar for all analysis so it is appended to the first few lines. Line 12-16 defines the material properties in ANSYS for an isotropic material. The creation of a volume and meshing it as one element is commanded by lines 18-24. Line 26-31 consists of the initiation of the modal analysis, followed by the output section in line 33-39. The commands for exporting the matrices is shown in line 37-38.

A.2. MATLAB code

The implementation of the topology optimization method in MATLAB was built to be a clear and easily adaptable environment for new ideas and experiments. It could definitely be more efficient in terms of lines of codes and speed but that has not been the objective while coding. All functionalities are written as building blocks which are assembled in the main file. The 99-line code by Sigmund [Sigmund, 2001] and the 88-line version by Andreassen et al. [Andreassen et al., 2011] have been used as a starting point.

A.2.1. Main file

The main file starts of with some initialization of the MATLAB workspace and declaration of the data structures and plotting variables. These are followed by the plotting out and data export options. Next the material properties and geometry are loaded selected by running the ParamMaterial and Param-Geometry file. The material interpolation method is selected by running the ParamInterpolation file. Some of the plotting setting are loaded by running the ParamVisualisation file.

The geometry is meshed by running the CreateMesh file and plotted if necessary. Using the method described in Section A.1, the mass and stiffness matrices are generated by running the CreateElement file. The boundary conditions of the model are applied by running the BoundaryConditions file and can also be plotted if necessary.

Next up the filtering, objective and constraint setting are available in the Main file. The initial design densities, minimum design density and number of modes to compute can also be selected in this part. Following this section, the eigenmodes and frequencies of the initial setup are computed using the FuncComputeFM file, the eigenmodes can be plotted or animated if necessary. Using the information of this, the reference mode shapes and eigenvalues are set.

The optimization loop follows next, starting with the initialization of the optimization parameters written in the ParamOptimization file. Some setting for continuation methods are available before the optimization loop is initiated.

The optimization loop starts out with computing the objective function values and gradients by using the OptObj function file. This is followed by the computation of the constraint values and the respective gradients by the OptConstr function file. The obtained information is fed to the mmasub file, the MMA optimization algorithm. The design density values are updated subsequently. The optimization process is printed in the MATLAB command window and plotted if selected. The continuation methods are applied in the last part of the optimization loop. A movie of the optimization process is created if selected. The post processing section starts of with processing the last step of the optimizer if density filtering is applied. Followed by a computation of the eigenmodes and values of the final step. The optimal result can be plotted if necessary. All of the data can be exported to a single file, for later processing or checking the optimization settings.

A.2.2. Parameter files

ParamMaterial

Parameter file containing the variables for:
- Young's modulus
- Poisson's ratio
- Density
- Shear modulus

ParamGeometry

Parameter file containing the variables for:
- Outer dimensions of the design domain.
- Solid rectangular or circular features.

ParamVisualization

Parameter file containing the variables for:
- Eigenmode plotting scales.
- Number of eigenmodes to plot in one figure.
- Animation of eigenmodes option.
- Optimization process recording initialization.

ParamInterpolation

Parameter file containing the variables for:
- All different interpolation methods shown in this thesis can be selected in the file.
- Penalization factors
- Initialization of the material interpolation functions and their derivatives.
- Plotting of the interpolation functions.

ParamFilter

Parameter file containing the variables for:
- Initialization of the filter matrices for fast multiplication.
- Computation of the filter matrices.
- Initialization of the heaviside transformation functions and their derivatives.

ParamOptimization

Parameter file containing the variables for:
- Enabling the plotting of the optimization history.
- Enabling the recording of the optimization history.
- Initialization of the optimization history data structures.
- Maximum number of iterations.
- Initialization of the MMA algorithm settings.

A.2.3. CreateMesh

This is the function file containing the mesh parameters for the number of nodes in x,y and z direction. This function creates a list of elements with their corresponding nodes, degrees of freedom numbers and location of the center. This is followed by creating a list of coordinates for a every node. Using the list of locations of the centers, the elements that remain solid throughout the optimization process and separated from the free elements. Finally the mesh can be plotted if required.

A.2.4. CreateElement

This is the function file that writes the full text file to be supplied to ANSYS, the procedure in this text file is described in Section A.1. It starts of by writing the material properties, dimensions and element type in the text file, followed by appending the analysis part. Next the ANSYS executable is called and the text file is supplied as input. The matrices are imported using the hb to msm MATLAB function file written by John Burkadt. As these are only the upper triangular part of the matrices, they are converted to a full matrix and stored in the Element data structure.

A.2.5. BoundaryConditions

Depending on the selected element type, different options for the boundary conditions are available. For example each of the sides of the planar element model can be fixed, specific nodes can also be fixed by specifying their location and directions to be fixed. Points masses can also be added to the model in this file by specifying their location and mass. Next the fixed and loaded DOF's of the model are found and written in the mesh data structure. The boundary conditions are plotted if necessary.

A.2.6. OptObj

The objective function file starts of processing the density filter in case it is enabled, followed by the computation of the eigenmodes and values using the FuncComputeFM file. The obtained eigenmodes and values and references are supplied to the mode tracking algorithm in the TrackModesFmin file. The reference eigenmodes and values are updated subsequently if required. Next the mean eigenvalue objective value is computed and the corresponding gradient values using the SensFreq function file. Followed by processing of the density or sensitivity filters.

A.2.7. FuncComputeFM

Within this function file the global mass and stiffness matrix is assembled, point masses are added and the first n eigenmodes and values are computed using MATLAB's eigs function for a subset of eigenvalues. The mode shape properties can be plotted if necessary. At this point the local and structural eigenmodes are sorted based on their modal mass. Finally the static load case is computed if required.

A.2.8. SensFreq

This is the function file which computes the sensitivity values of each elements with respect a single eigenvalue. For every combination of eigenmode and value, a gradient vector is returned. The sensitivities with respect to the static results are computed by supplying it with a zero eigenvalue.

A.2.9. TrackModesFmin

In this function file the algorithm described in Chapter 4 is implemented. The eigensolver output is supplied along with the reference information and the corresponding order of eigenmodes and values is available as output.

A.2.10. OptConstr

Within this function file all constraint values are computed with their corresponding gradient vectors. At the end of the file the filters are processed.

A.2.11. Plotting functions

Different plotting functions are available to the user, each of them has an export function

PlotHistory
Plotting function which is implemented in the optimization loop. Is able to plot different information of the optimization process such as:
  • Eigenfrequency history
  • Optimizer step size
  • Measure of non-discreteness history
  • Current design density field
  • Objective gradient field
  • Constraint values
  • Constraint gradient field
  • Mode shapes

PlotMesh
Plotting function for the generated mesh, can be plotted with or without boundary conditions and a design density field.

PlotModes
Plotting function for mode shapes, these are either plotted or animated.

PlotModesSet
Uses the PlotModes function to generate a set of plots, separated into structural and local eigenmodes.

PlotModeProp
Plotting function for the mode shape properties defined in Section 3.1.1.

Material interpolation graphs

B.1. Existing methods

B.1.1. Power law

Power law interpolation method first introduced by Rozvany et al. [Rozvany et al., 1992].

$$f_k(\rho) = \rho^{P_k} \qquad f_m(\rho) = \rho^{P_m} \qquad \rho_{min} \leq \rho \leq 1 \tag{B.1}$$
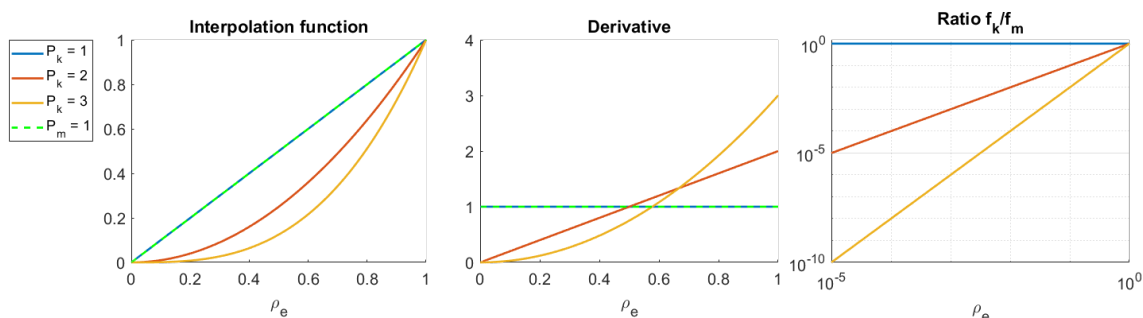

Figure B.1: Powerlaw interpolation function with different penalization factors for stiffness interpolation.

Local eigenmode prevention technique originally introduced for the power law interpolation method by Pedersen [Pedersen, 2000]. Can be applied on any material interpolation method.

$$f_k(\rho) = \epsilon + \rho^{P_k} \cdot (1-\epsilon) \qquad f_m(\rho) = \epsilon + \rho^{P_m} \cdot (1-\epsilon) \qquad 0 \leq \rho \leq 1 \tag{B.2}$$
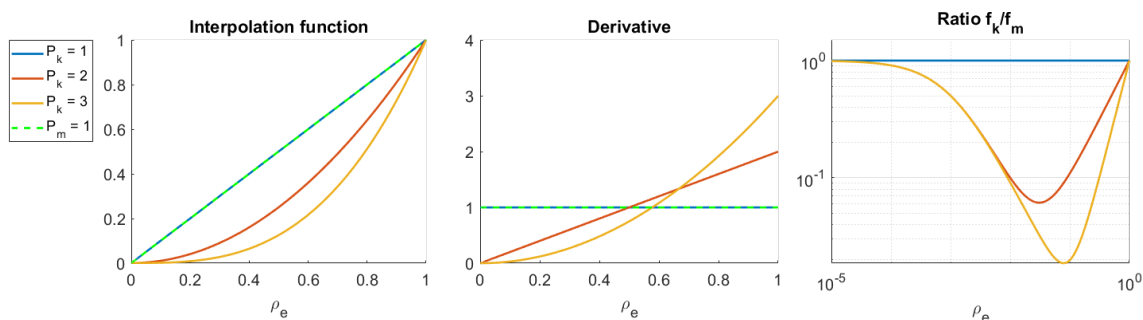

Figure B.2: Powerlaw interpolation function with Pedersen's modification with different penalization factors for stiffness interpolation.

B.1.2. RAMP

An alternative to the power law interpolation functions was introduced by Stolpe and Svanberg [Stolpe and Svanberg, 2001], which is now well known as the RAMP method.

$$f_k(\rho) = \frac{\rho}{1 + P_k(1-\rho)} \qquad f_m(\rho) = \rho^{P_m} \qquad \rho_{min} \leq \rho \leq 1 \tag{B.3}$$

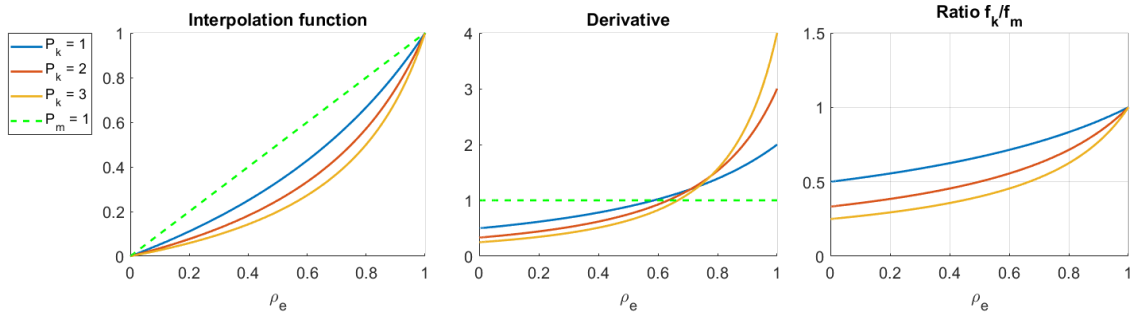**Figure B.3:** RAMP interpolation function with different penalization factors for stiffness interpolation.

B.1.3. Polynomial

A different take on the local eigenmode problem with a new interpolation method was presented by Zhu et al. [Zhu et al., 2010].

$$f_k(\rho) = r\rho^{P_k} + (1-r)\rho \qquad f_m(\rho) = \rho^{P_m} \qquad \rho_{min} \leq \rho \leq 1 \tag{B.4}$$
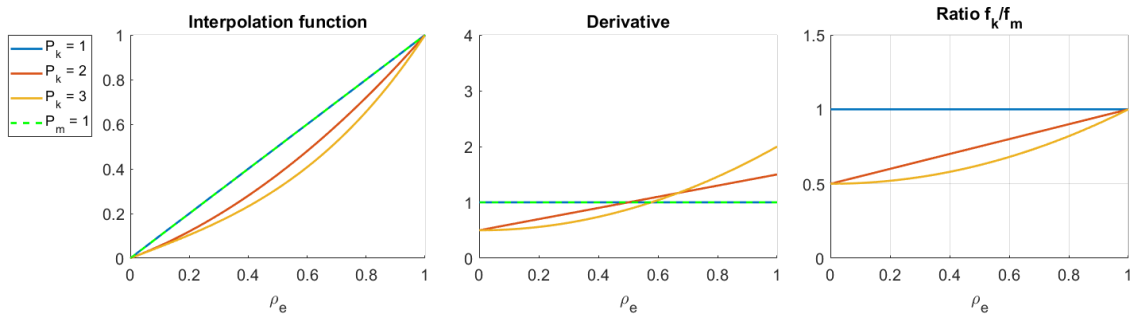

Figure B.4: Polynomial interpolation function with different penalization factors for stiffness interpolation.

B.1.4. SINH

Another alternative to the power law interpolation functions was introduced by Bruns [Bruns, 2005].

$$f_k(\rho) = \frac{\sinh(P_k\rho)}{\sinh(P_k)} \qquad f_m(\rho) = 1 - \frac{\sinh(P_m(1-\rho))}{\sinh(P_m)} \qquad \rho_{min} \le \rho \le 1 \tag{B.5}$$
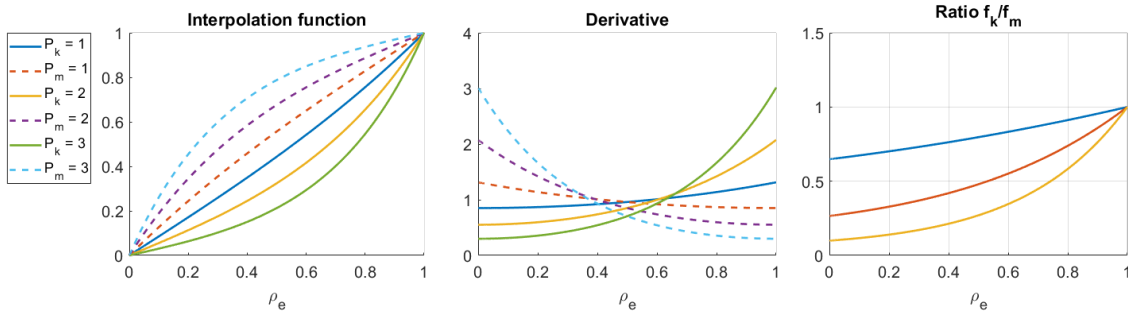

Figure B.5: SINH interpolation function with different penalization factors for stiffness and mass

B.2. New methods

B.2.1. Full sine

New method developed in this thesis work, using the linear interpolation method as a basis and adding a full sine shape function to the derivative.

$$f_k(\rho) = \rho - \frac{P_k}{2\pi}(1-\cos(2\pi\rho)) \qquad f_m(\rho) = \rho + \frac{P_m}{2\pi}(1-\cos(2\pi\rho)) \qquad \rho_{min} \le \rho \le 1 \tag{B.6}$$
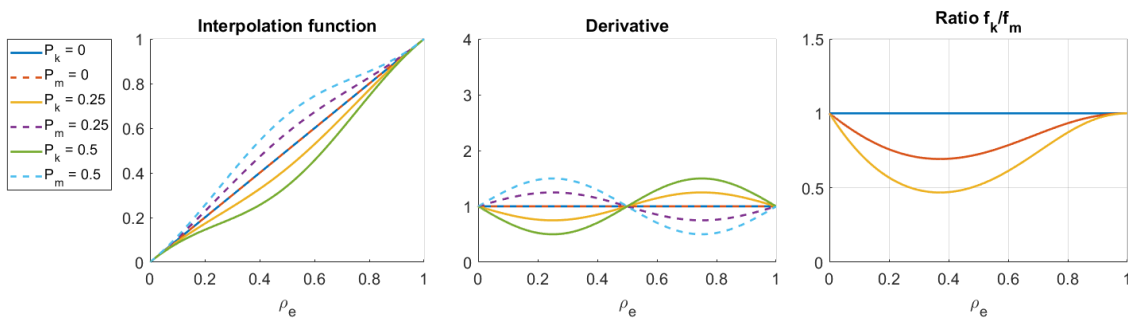

Figure B.6: Full sine interpolation function with different values for the sine amplitude of both the stiffness and mass interpolation.

B.2.2. Half cosine

New method developed in this thesis work, using the linear interpolation method as a basis and adding a half a cosine shape function to the derivative.

$$f_k(\rho) = \rho - \frac{P_k}{\pi}\sin(\pi\rho) \qquad f_m(\rho) = \rho + \frac{P_m}{\pi}\sin(\pi\rho) \qquad \rho_{min} \leq \rho \leq 1 \tag{B.7}$$
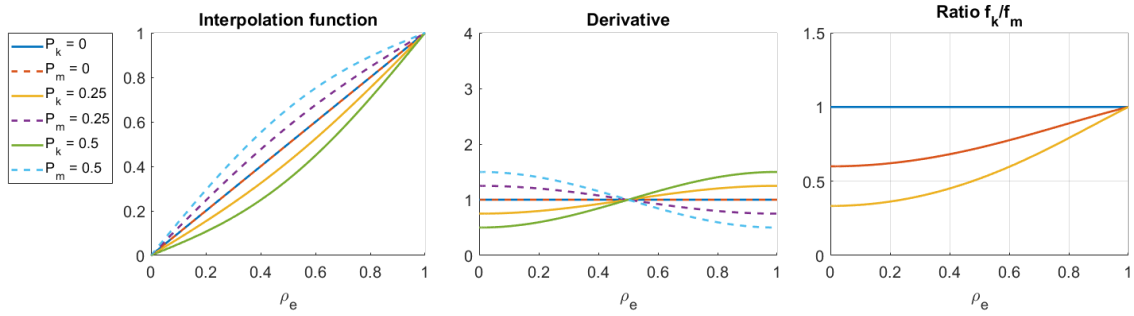

Figure B.7: Half sine interpolation function with different values for the sine amplitude of both the stiffness and mass interpolation.

B.2.3. Linear derivative

New method developed in this thesis work, using the linear interpolation method as a basis and adding a linear shape function to the derivative.

$$f_k = P_k\rho^2 + (1-P_k)\rho \qquad f_m = -P_m\rho^2 + (1+P_m)\rho \qquad \rho_{min} \leq \rho \leq 1 \qquad 0 \leq P_k, P_m \leq 1 \tag{B.8}$$
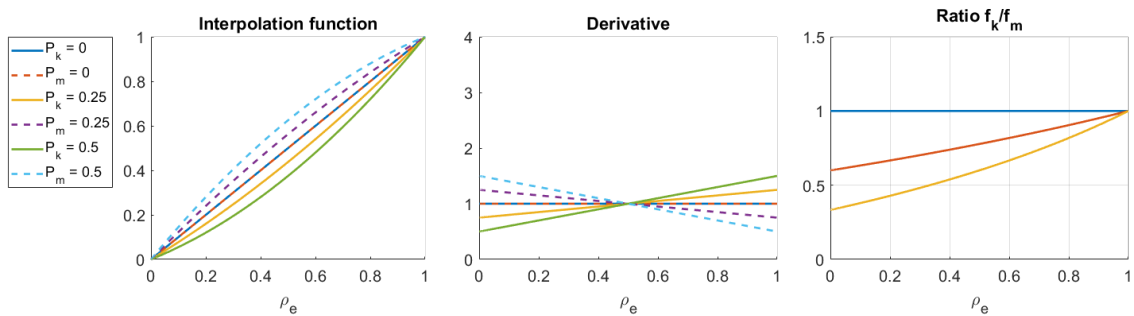

Figure B.8: Linear derivative interpolation function with different values for the slope of the derivative for both the stiffness and mass interpolation.

Sensitivity analysis of objective functions

C.1. Eigenvalue sensitivity analysis

In this section the analytical sensitivity analysis of a single eigenvalue is shown for the symmetric positive definite eigenvalue problem shown in Equation C.1. This analysis follows the method described by Tortorelli and Michaleris [Tortorelli and Michaleris, 1994]. The eigenmodes resulting from the eigenvalue problem are mass normalized using Equation C.2.

$$\left[K(\rho) - \lambda_j(\rho)M(\rho)\right]\Phi_j(\rho) = 0 \qquad j = 1, ..., n \tag{C.1}$$

$$\Phi_i^T M \Phi_j = \delta_{ij} \tag{C.2}$$

Direct differentiation of the eigenvalue problem to the design density leads to:

$$\left[\frac{\partial K(\rho)}{\partial \rho} - \lambda_j(\rho)\frac{\partial M(\rho)}{\partial \rho}\right]\Phi_j(\rho) - \frac{\partial \lambda_j(\rho)}{\partial \rho}M(\rho)\Phi_j(\rho) + \left[K(\rho) - \lambda_j(\rho)M(\rho)\right]\frac{\partial \Phi_j(\rho)}{\partial \rho} = 0 \tag{C.3}$$

Premultiplying by an arbitrary vector γ^T and rearranging the following is obtained:

$$\gamma^T M(\rho)\Phi_j(\rho)\frac{\partial \lambda_j(\rho)}{\partial \rho} = \gamma^T\left[\frac{\partial K(\rho)}{\partial \rho} - \lambda_j(\rho)\frac{\partial M(\rho)}{\partial \rho}\right]\Phi_j(\rho) + \gamma^T\left[K(\rho) - \lambda_j(\rho)M(\rho)\right]\frac{\partial \Phi_j(\rho)}{\partial \rho} = 0 \tag{C.4}$$

Now γ is replaced with Φ_j since γ is arbitrary, resulting in:

$$\Phi_j^T(\rho)M(\rho)\Phi_j(\rho)\frac{\partial \lambda_j(\rho)}{\partial \rho} = \Phi_j^T(\rho)\left[\frac{\partial K(\rho)}{\partial \rho} - \lambda_j(\rho)\frac{\partial M(\rho)}{\partial \rho}\right]\Phi_j(\rho) + \Phi_j^T(\rho)\left[K(\rho) - \lambda_j(\rho)M(\rho)\right]\frac{\partial \Phi_j(\rho)}{\partial \rho}$$
$$\tag{C.5}$$

The left-hand side of this reduced to the eigenvalue sensitivity by substituting in the Equation C.2 for the mass normalization. The term containing the eigenmode derivative is canceled out by using the identity $cAb = bA^Tc$, the symmetry of eigenvalue problem and equation C.1 for the eigenvalue problem. This results in:

$$\frac{\partial \Phi_j(\rho)}{\partial \rho}^T\left[K(\rho) - \lambda_j(\rho)M(\rho)\right]\Phi_j(\rho) = 0 \tag{C.6}$$

What remains is an explicit sensitivity expression for the eigenvalues.

$$\frac{\partial \lambda_j(\rho)}{\partial \rho} = \Phi_j^T(\rho)\left[\frac{\partial K(\rho)}{\partial \rho} - \lambda_j(\rho)\frac{\partial M(\rho)}{\partial \rho}\right]\Phi_j(\rho) \tag{C.7}$$

Using Equation C.8 this expression can be evaluated for every element as shown in Equation C.9. In this equation k_e and m_e are the element mass and stiffness matrices.

$$K(\rho) = \sum_{e=1}^{n} f_k(\rho_e) \cdot k_e \qquad M(\rho) = \sum_{e=1}^{n} f_m(\rho_e) \cdot m_e \tag{C.8}$$

$$\frac{\partial \lambda_j(\rho_e)}{\partial \rho_e} = \Phi_j^T(\rho_e) \left[\frac{\partial f_k(\rho_e)}{\partial \rho_e} k_e - \lambda_j(\rho_e) \frac{\partial f_m(\rho_e)}{\partial \rho_e} m_e \right] \Phi_j(\rho_e) \tag{C.9}$$

C.2. Mean eigenvalue sensitivity analysis

The mean eigenvalue shown in Equation C.10 and introduced by Ma et al. [Ma et al., 1994] is a weighted sum of individual eigenvalues, the sensitivity analysis for a single eigenvalue has been shown in the previous section. Therefore the sensitivity analysis of the mean eigenvalue is again a sum of the sensitivity information of the individual eigenvalues.

$$\lambda^*(\rho) = \begin{cases} \lambda_t^* + \left(\dfrac{1}{\sum_{i=1}^{m} w_i} \sum_{i=1}^{m} w_i (\lambda_i - \lambda_{t_i})^n \right)^{1/n} & \text{for} \quad n = \pm 1, \pm 2, \ldots n \neq 0 \\[4mm] \lambda_t^* + \exp\left(\dfrac{1}{\sum_{i=1}^{m} w_i} \sum_{i=1}^{m} w_i \ln |\lambda_i - \lambda_{t_i}| \right) & \text{for} \quad n = 0 \end{cases} \tag{C.10}$$

The sensitivity analysis of this objective functions results in:

$$\frac{\partial \lambda^*(\rho_e)}{\partial \rho_e} = \begin{cases} \dfrac{(\lambda^* - \lambda_t^*)^{1-n}}{\sum_{i=1}^{m} w_i} \sum_{i=1}^{m} \dfrac{w_i}{(\lambda_i - \lambda_{t_i})^{1-n}} \dfrac{\partial \lambda_i(\rho_e)}{\partial \rho_e} & \text{for} \quad n = \pm 1, \pm 2, \ldots n \neq 0 \\[4mm] \dfrac{|\lambda^* - \lambda_t^*|}{\sum_{i=1}^{m} w_i} \sum_{i=1}^{m} \dfrac{w_i}{|\lambda_i - \lambda_{t_i}|} \dfrac{\partial \lambda_i(\rho_e)}{\partial \rho_e} & \text{for} \quad n = 0 \end{cases} \tag{C.11}$$

C.3. Discreteness objective sensitivity analysis

The discreteness objective function in Equation C.12 uses the smooth or non-smooth discreteness value shown in Equation C.13 and C.14 to modify the objective function. The sensitivity analysis of this new objective function is done using the chain rule on the objective function. The original objective function can in this case be a single eigenvalue or the mean eigenvalue for which the sensitivity analysis has already been presented in the previous sections. The derivatives of the SD and NSD values are obtained by straightforward differentiation and are shown in Equation C.15 and C.16.

$$f_{obj,c} = (N)SD(\rho)^{\alpha} \cdot f_{obj} \tag{C.12}$$

$$SD(\rho) = \frac{\sum_{e}^{n} 4(\rho_e - 0.5)^2}{n} \qquad 0 \leq \rho_e \leq 1 \tag{C.13}$$

$$NSD(\rho) = \frac{\sum_{e}^{n} 2|\rho_e - 0.5|}{n} \qquad 0 \leq \rho_e \leq 1 \tag{C.14}$$

$$\frac{\partial SD(\rho_e)}{\partial \rho_e} = \frac{8(\rho_e - 0.5)}{n} \tag{C.15}$$

$$\frac{\partial NSD(\rho_e)}{\partial \rho_e} = \frac{2(\rho_e - 0.5)}{n|\rho_e - 0.5|} \tag{C.16}$$

Combining these results by using the chain rule results in:

$$\frac{\partial f_{obj,c}}{\partial \rho_e} = (N)SD(\rho_e)^{\alpha-1}\left(\alpha f_{obj}\frac{\partial (N)SD(\rho_e)}{\partial \rho_e} + (N)SD(\rho_e)\frac{\partial f_{obj}(\rho_e)}{\partial \rho_e}\right) \tag{C.17}$$

D

Sensitivity analysis of constraints

D.1. Volume constraint

To constrain the amount of material used in the design domain, a volume constraint can be applied. This is the sum of the volume of each element v_e multiplied by its design density value ρ_e. A relative upper limit of V_{max} and a lower limit of V_{min} can be specified. The volume constraint shown in Equation D.1. The sensitivity analysis of these constraints is conducted using direct differentiation, resulting in Equations D.2 and D.3.

$$\frac{\sum_{e=1}^{n} \rho_e v_e}{V \cdot V_{max}} - 1 < 0 \qquad 1 - \frac{\sum_{e=1}^{n} \rho_e v_e}{V \cdot V_{min}} < 0 \tag{D.1}$$

$$\frac{\partial C_{Vmax}}{\partial \rho_e} = \frac{v_e}{V \cdot V_{max}} \tag{D.2}$$

$$\frac{\partial C_{Vmin}}{\partial \rho_e} = \frac{v_e}{V \cdot V_{min}} \tag{D.3}$$

D.2. Frequency ratio constraint

The eigenfrequency ratio r_{ij} as defined in Equation D.4 can be constrained by using the two constraints shown in D.5. The sensitivity analysis of this constraint is conducted using direct differentiation and the eigenvalue sensitivity analysis for a single eigenvalue, resulting in Equation D.6.

$$\frac{\lambda_i}{\lambda_j} = r_{ij} \qquad \frac{\omega_i^2}{\omega_j^2} = r_{ij}^2 \tag{D.4}$$

$$\frac{\lambda_i}{\lambda_j r_{ij}} - 1 < 0 \qquad 1 - \frac{\lambda_i}{\lambda_j r_{ij}} < 0 \tag{D.5}$$

$$\frac{\partial C_{fr}}{\partial \rho_e} = (-)\frac{\lambda_j \frac{\partial \lambda_i}{\partial \rho_e} - \lambda_i \frac{\partial \lambda_j}{\partial \rho_e}}{\lambda_j^2 r_{ij}} \tag{D.6}$$

D.3. Static compliance constraint

When optimizing for eigenfrequencies of a structure the result in some cases can be a relatively weak structure when loaded statically. This is why in some cases a constraint on the maximum static compliance of a structure is applied. A load vector F_i can be used to find the displacement vector u using Equation D.7. The compliance C_i corresponding to load vector F_i can be found using Equation D.9. A maximum compliance C_{max} can be enforced using the constraint shown in D.10. The sensitivity analysis of the compliance value is conducted using direct differentiation with respect to the design density resulting in Equation D.11. In which u_e is the resulting displacement vector corresponding to element e.

$$Ku = F_i \tag{D.7}$$

$$K(\rho) = \sum_{e=1}^{n} f_k(\rho_e) \cdot k_e \qquad \frac{\partial K(\rho_e)}{\partial \rho_e} = \frac{\partial f_k(\rho_e)}{\partial \rho_e} k_e \tag{D.8}$$

$$C_i = F_i^T u \qquad C_i = u^T K u \tag{D.9}$$

$$\frac{C_i}{C_{max}} - 1 < 0 \tag{D.10}$$

$$\frac{\partial C_{comp}}{\partial \rho_e} = \frac{u_e^T \frac{\partial f_k(\rho_e)}{\partial \rho_e} k_e u_e}{C_{max}} \tag{D.11}$$

D.4. Stiffness constraint

Instead of the compliance of the total structure with respect to a load vector, the stiffness of a structure in a certain direction at a certain location can be constrained in a similar way. By using a load vector F_i with only a unit force on the DOF of interest, resulting in displacement u_i of this DOF. The stiffness k_i of this DOF can then be evaluated using Equation D.13. A constraint on the minimum stiffness of DOF i can be applied using the constraint in D.14. The sensitivity analysis of this constraint is similar to the static compliance stiffness analysis and results in Equation D.15.

$$u_i = F_i^t u \qquad u_i = u^T K u \tag{D.12}$$

$$k_i = \frac{1}{u_i} \tag{D.13}$$

$$\frac{k_{min}}{k_i} - 1 < 0 \tag{D.14}$$

$$\frac{\partial C_{stiff}}{\partial \rho_e} = k_{min} u_e^T \frac{\partial f_k(\rho_e)}{\partial \rho_e} k_e u_e \tag{D.15}$$

D.5. SRV constraint

The sum of reciprocal variables (SRV) constraint was introduced by Fuchs et al. [Fuchs et al., 2005] as a method to push the design towards a discrete final result. The SRV value can be computed using Equation D.16. With a specified volume for the design domain this value can also be computed for a discrete design. Together with the computed discrete value, a constraint can be formed as shown in Equation D.17. For numerical reasons a relaxation parameter η is added. The downside of this method is the required specific volume. The sensitivity analysis of this constraint is conducted using direct differentiation with respect to the design densities, resulting in Equation D.18.

$$SRV(\rho) = \sum_{e}^{n} \frac{1}{\rho_e} \qquad \rho_{min} \leq \rho_e \leq 1 \tag{D.16}$$

$$1 - \frac{SRV(\rho)}{\eta SRV_{discrete}(V)} < 0 \tag{D.17}$$

$$\frac{\partial C_{srv}}{\partial \rho_e} = \frac{1}{\rho_e^2 \eta SRV_{discrete}(V)} \tag{D.18}$$

D.6. Discreteness constraint

The discreteness constraint based on the smooth (SD) or non-smooth (NSD) is shown in Equation D.19. The sensitivity analysis of this constraint is conducted by direct differentiation with respect to the design densities, resulting in Equations D.22 and D.23.

$$\eta - (N)SD(\rho) < 0 \tag{D.19}$$

$$SD(\rho) = \frac{\sum_{e}^{n} 4(\rho_e - 0.5)^2}{n} \qquad 0 \leq \rho_e \leq 1 \tag{D.20}$$

$$NSD(\rho) = \frac{\sum_{e}^{n} 2|\rho_e - 0.5|}{n} \qquad 0 \leq \rho_e \leq 1 \tag{D.21}$$

$$\frac{\partial C_{SD}(\rho_e)}{\partial \rho_e} = -\frac{8(\rho_e - 0.5)}{n} \tag{D.22}$$

$$\frac{\partial C_{NSD}(\rho_e)}{\partial \rho_e} = -\frac{2(\rho_e - 0.5)}{n|\rho_e - 0.5|} \tag{D.23}$$

E
Sensitivity analysis of multiple eigenvalues

In this appendix the sensitivity analysis of multiple eigenvalues according to Seyranian et al. [Seyranian et al., 1994] is shown. This analysis is used in the algorithm presented by Du and Olhoff [Du and Olhoff, 2007] which is able to optimize a structure exhibiting multiplicity of eigenvalues. Assuming that the eigenvalue problems shown in Equation E.1 results in an N-fold multiple of eigenvalue $\tilde{\lambda}$, numbered 1 to N.

$$[K(\rho) - \lambda_j(\rho)M(\rho)]\Phi_j(\rho) = 0 \qquad j = 1, ..., n \tag{E.1}$$

$$\tilde{\lambda} = \lambda_j = \omega_j^2, \qquad j = 1, ..., N \tag{E.2}$$

The sensitivity analysis of a multiple eigenvalue is not straight forward due to the fact that the eigenmodes of the repeated eigenvalues are not unique, any linear combination of the eigenmodes will satisfy the original eigenvalue problem. The eigenmodes are assumed to be mass normalized for this analysis. Following the sensitivity analysis, a linear combination of the eigenmodes is introduced.

$$\tilde{\Phi}_j = \sum_{k=1}^{N} \beta_{jk}\Phi_k, \qquad j = 1, ..., N \tag{E.3}$$

$$\Phi_i^T M \Phi_j = \delta_{ij} \tag{E.4}$$

In which β_{jk} are unknown coefficients to be determined. The sensitivities of multiple eigenvalues can be found as shown by Courant and Hilbert [Courant, 1953], Wittrick and Lancaster [Lancaster, 1964] by solving a subeigenvalue problem. Considering the change $\Delta\rho_e$ of one of the design variables, the mass and stiffness matrices change respectively by:

$$K = K + \frac{\partial K}{\partial \rho_e}\Delta\rho_e \tag{E.5}$$

$$M = M + \frac{\partial M}{\partial \rho_e}\Delta P_e \tag{E.6}$$

Using the same expansion for the eigenvalue and eigenvector:

$$\lambda_j(\rho_e + \epsilon\Delta\rho_e) = \tilde{\lambda} + \epsilon\mu_j(\rho_e, \epsilon\Delta\rho_e) + o(\epsilon), \qquad j = 1, ..., N \tag{E.7}$$

$$\Phi_j(\rho_e + \epsilon\Delta\rho_e) = \tilde{\Phi}_j + \epsilon\nu_j(\rho_e + \epsilon\Delta\rho_e) + o(\epsilon), \qquad j = 1, ..., N \tag{E.8}$$

In these equations μ_j and v_j are the unknown eigenvalue and eigenvector sensitivities. Substituting this result into the eigenvalue problem leads to:

$$\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\tilde{\Phi}_j + \left(K - \tilde{\lambda}M\right)v_j = \mu_j M\tilde{\Phi}_j \tag{E.9}$$

Premultiplying the result with one of the multiple eigenvectors Φ_s^T gives

$$\Phi_s^T\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\tilde{\Phi}_j + \Phi_s^T\left(K - \tilde{\lambda}M\right)v_j = \mu_j\Phi_s^T M\tilde{\Phi}_j, \qquad s = 1,...,N \tag{E.10}$$

The right term on the left hand side of this equation drops out as Φ_s^T is an eigenmode corresponding to the multiple eigenvalue $\tilde{\lambda}$. What is left is the following equation:

$$\Phi_s^T\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\tilde{\Phi}_j = \mu_j\delta_{sj}, \qquad s = 1,...,N \tag{E.11}$$

Now substituting relationship for the linear combination of eigenmodes $\tilde{\Phi}_j$ results in a system of linear algebraic equations of unknown coefficients β_{jk}.

$$\sum_{k=1}^{N}\beta_{jk}\left[\Phi_s^T\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\Phi_k - \mu_j\delta_{sk}\right] = 0, \qquad s = 1,...,N \tag{E.12}$$

A nontrivial solution to these equations only exists if the determinant of the system is equal to zero

$$\det\left[\Phi_s^T\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\Phi_k - \mu_j\delta_{sk}\right] = 0, \qquad s,k = 1,...,N \tag{E.13}$$

This is the equation used to determine the sensitivities of the multiple eigenvalues, however it is only valid for the change of a single design variable. If the off-diagonal terms of matrix of dimension N are zero, the eigenvalue sensitivities are equal to the sensitivities found using the sensitivity analysis for a single eigenvalue. The analysis for multiple design changes is similar and results in:

$$\det\left[\sum_{e=1}^{n}\Phi_s^T\left(\frac{\partial K}{\partial \rho_e} - \tilde{\lambda}\frac{\partial M}{\partial \rho_e}\right)\Phi_k e - \mu_j\delta_{sk}\right] = 0, \qquad s,k = 1,...,N, \qquad e = 1,...,n \tag{E.14}$$

In which e is any vector of arbitrary design changes with a unit norm. This equation can be simplified by introducing the generalized gradient vectors f_{sk} of dimension n. The subscripts of the generalized gradient vectors denote the eigenmodes from which they are computed.

$$f_{sk} = \left(\Phi_s^T\left(\frac{\partial K}{\partial \rho_1} - \tilde{\lambda}\frac{\partial M}{\partial \rho_1}\right)\Phi_k,...,\Phi_s^T\left(\frac{\partial K}{\partial \rho_n} - \tilde{\lambda}\frac{\partial M}{\partial \rho_n}\right)\Phi_k\right) \tag{E.15}$$

Using the formulation for the generalized gradient vectors, the equation is simplified to:

$$\det\left[f_{sk}^T\Delta\rho - \Delta\lambda\delta_{sk}\right] = 0, \qquad s,k = 1,...,N \tag{E.16}$$

The change of eigenvalues in the N-fold multiple eigenvalue can be predicted using this equation for any arbitrary step $\Delta\rho$ in the design variables. The method developed by Du and Olhoff relies on using this equation to find the optimal step in design densities.