UNIVERSITE DE LIEGE

Aerospace and Mechanical Engineering Department

Topology Optimization of Electrostatic MEMS Including Stability Constraints

PhD Thesis Dissertation
submitted in partial fulfillment
of the requirements for the degree of
'Docteur en Sciences de l'Ingénieur'

by

Etienne Lemaire
Ingénieur Civil Electro-Mécanicien (Aérospatiale)

April 2013

Members of the examination committee

Jean-Claude Golinval (President of the committee)
Professor - University of Liège

Pierre Duysinx (Supervisor)
Professor - University of Liège

Eric Béchet
Professor - University of Liège

Vincent Denoël
Professor - University of Liège

Patrick Dular
Research Director F.R.S.-F.N.R.S. - University of Liège

Tristan Gilet
Professor - University of Liège

Fred van Keulen
Professor - Delft University of Technology (Netherlands)

Véronique Rochus
PhD, MEMS Design Engineer - IMEC (Belgium)

Ole Sigmund
Professor - Technical University of Denmark

Abstract

Among actuation techniques available for MEMS devices, electrostatic actuation is often used as it provides a short response time and is relatively easy to implement. However, these actuators possess a limit voltage called pull-in voltage beyond which they are unstable. The pull-in effect, can eventually damage the device since it can be impossible to separate the electrodes afterward. Consequently, pull-in phenomenon should be taken into account during the design process of electromechanical microdevices to ensure that it is avoided within utilization range. In this thesis, a topology optimization procedure which allows controlling pull-in phenomenon during the design process is developed.

A first approach is based on a simplified optimization problem where the optimization domain is separated from the electrical domain by a perfectly conducting material layer making the optimization domain purely mechanical. This assumption reduces the difficulty of the optimization problem as the location of the electrostatic forces is then independent from the design. However, it allows us to develop and validate a design function based on pull-in voltage in the framework of a topology optimization problem.

Nevertheless, in some applications, the developed pull-in voltage optimization procedure suffers from design oscillations that prevent from reaching solution. In order to solve this issue, we propose to investigate an alternative approach consisting in formulating a linear eigenproblem approximation for the nonlinear stability problem. The first eigenmode of the proposed stability eigenproblem corresponds to the actual pull-in mode while higher order modes allow estimating upcoming instability modes. By including several instability modes into a multiobjective formulation, it is possible to circumvent the oscillations encountered with pull-in voltage design function.

Next, the possibility to generalize the pull-in optimization problem by removing the separation between optimization and electrostatic domains is studied. Unlike the original method, the dielectric permittivity has then to depend on the pseudo-density like the Young Modulus to represent the different electrostatic behavior of void and solid. Additionally, in order to render perfect conducting behavior for the structural part of the optimization domain, a fictitious permittivity is also introduced into the material model. Difficulties caused by non-physical local instability modes could be solved by using a force filtering technique which removes electrostatic forces originating from numerical inaccuracies of the modeling method. Thanks to these improvements, the optimization problem based on the pull-in design function can be generalized. As a result, the optimizer is able to adapt the electrostatic force distribution applied on the structure which leads to a higher efficiency of the optimal device.

In order to illustrate the interest of the pull-in voltage design function, the pull-in voltage optimization problem is merged with the electrostatic actuator optimization problem. In this new optimization problem, the pull-in voltage does not appear anymore in the objective function but in a constraint which prevents the pull-in voltage to decrease below a given minimal value. Firstly, the new optimization problem is compared to the basic electrostatic actuator design procedure on basis of a numerical application. The pull-in voltage constraint proved to be very useful since it prevents the pull-in voltage of the mechanism to decrease below the driving voltage during the optimization process. Finally, the effect of geometric nonlinearity modeling is also tested on numerical applications of our optimization procedure.

Acknowledgements

I gratefully acknowledge the project ARC MEMS, Action de Recherche Concertée 03/08-298 funded by the Communauté Française de Belgique, which supported part of this thesis.

I express sincere gratitude to my advisor Professor Pierre Duysinx for offering me the opportunity to work in his research group and to complete this thesis. Along these years, I could always count on his unwavering support, guidance, advice and cheerfulness. I also wish to thank him for all the time he spent reading and correcting this thesis.

Many thanks also to the present and former colleagues from the Aerospace and Mechanical Engineering department, who guided me in my research, helped me in my teaching assistant work, took part to the correction of this manuscript, or simply participated to the excellent atmosphere. I am especially thankful to my former office mate Laurent Van Miegroet for pleasant fellowship along these years and to Véronique Rochus for all the support and advice about electromechanical modeling.

I am grateful to Professor Mathias Stolpe for welcoming me at the Department of Mathematics of Technical University of Denmark during one month. This stay has been for me a chance to exchange many ideas with other members of the TopOpt group and a great source of inspiration for this thesis.

I would like to acknowledge Professors Eric Béchet, Vincent Denoël, Tristan Gilet, Jean-Claude Golinval, Fred van Keulen, Ole Sigmund, Research director Patrick Dular and Doctor Véronique Rochus for accepting to participate in the examination committee of this doctoral thesis.

Last but not least, I could not finish this thesis without the continuous support and encouragement from Julie, my parents, my sisters and my brother. They deserve more thanks than I can write.

Author Publications

Publication in refereed journal

- E. Lemaire, V. Rochus, J.-C. Golinval, and P. Duysinx. Microbeam pull-in voltage topology optimization including material deposition constraint. *Computer Methods in Applied Mechanics and Engineering*, 194:4040–4050, 2008.

Book chapter

- O. Brüls, E. Lemaire, P. Duysinx, and P. Eberhard. Optimization of multibody systems and their structural components. In K. Arczewski, W. Blajer, J. Fraczek, and M. Wojtyra, editors, *Multibody Dynamics*, volume 23 of *Computational Methods in Applied Sciences*, pages 49–68. Springer Netherlands, 2011.

Publications in proceedings

- E. Lemaire, P. Duysinx, V. Rochus, and J.-C. Golinval. Improvement of pull-in voltage of electromechanical microbeams using topology optimization. In *Proceedings of the III European Conference on Computational Mechanics*. June 2006.

- E. Lemaire, V. Rochus, C. Fleury, J.-C. Golinval, and P. Duysinx. Topology optimization of microbeams including layer deposition manufacturing constraints. In *Proceedings of the 7th World Congress on Structural and Multidisciplinary Optimization*. 2007.

- O. Brüls, E. Lemaire, P. Eberhard, and P. Duysinx. Design of mechanism components using topology optimization and flexible multibody simulation. In *Proceedings of the 8th World Congress on Computational Mechanics*. 2008.

- P. Duysinx, C. Fleury, L. Van Miegroet, E. Lemaire, O. Bruls, and M. Bruyneel. Stress constrained topology and shape optimization: Specific character and large scale optimization algorithms. In *Proceedings of the 8th World Congress on Computational Mechanics*. 2008.

- E. Lemaire, V. Rochus, J.-C. Golinval, and P. Duysinx. Electromechanical microdevice pull-in voltage maximization using topology optimization. In *Proceedings of the 8th. World Congress on Computational Mechanics*. 2008.

- E. Lemaire, P. Duysinx, and V. Rochus. Multiphysic topology optimization of electromechanical microdevices considering pull-in voltage using an electrostatic force filter. In *Proceedings of the 8th World Congress on Structural and Multidisciplinary Optimization*. June 2009.

- E. Lemaire, L. Van Miegroet, P. Duysinx, and V. Rochus. Influence of the material model on local pull-in in electromechanical microdevices topology optimization. In *PLATO-N International Workshop-Extended Abstracts*. 2009.

- E. Lemaire, L. Van Miegroet, T. Schoonjans, P. Duysinx, and V. Rochus. Multiphysic topology optimization of electromechanical micro-actuators considering pull-in effect. In *Proceedings of the IVth European Conference on Computational Mechanics*. 2010.

- V. Rochus, E. Lemaire, and C. Geuzaine. Dual approach for an accurate estimation of pull-in voltage. In *Proceedings of the IVth European Conference on Computational Mechanics*. 2010.

- E. Lemaire, L. Van Miegroet, V. Rochus, and P. Duysinx. Topology optimization of electrostatic micro-actuators including electromechanical stability constraint. In *Proceedings of the 5th International Conference on Advanced COmputational Methods in ENgineering*. 2011.

- E. Tromme, E. Lemaire, and P. Duysinx. Topology optimization of compliant mechanisms: Application to vehicle suspensions. In *Proceedings of the 5th International Conference on Advanced COmputational Methods in ENgineering*. 2011.

- E. Lemaire, L. Van Miegroet, E. Tromme, L. Noël, and P. Duysinx. Eigenproblem formulation for electromechanical microsystem pull-in voltage optimization. In *Proceedings of the 10th World Congress on Structural and Multidisciplinary Optimization*. May 2013.

Contents

List of symbols **1**

1 Introduction **3**

1.1 General context of the thesis 3

1.2 Objectives of the thesis 4

1.3 Layout of the thesis 5

2 Electromechanical modeling **9**

2.1 Introduction 9

2.2 Electrostatic microsystems 10

2.2.1 RF switches 10

2.2.2 Microelectromechanical resonator 12

2.2.3 Microelectromechanical gyroscopes 14

2.2.4 Electrostatically actuated micromirrors 14

2.2.5 Microlens optical scanner 16

2.3 Insight of the behavior of the electrostatic actuator 17

2.4 Electromechanical finite element formulation 20

2.4.1 Variational approach 20

2.4.2 Tangent stiffness matrix 22

2.4.3 Summary 23

2.5 Normal flow algorithm 24

2.5.1 Homotopy methods 24

2.5.2 Normal flow algorithm mathematical formulation 26

2.5.3 Implementation of the normal flow algorithm for electromechanical problems. 27

2.6 Numerical application 34

2.6.1 Benchmark presentation 34
2.6.2 Normal flow results 34
2.6.3 Comparison with approximate closed-form expression 35
2.7 Conclusion 36

3 Optimization of multiphysic microsystems 37
3.1 Introduction 37
3.2 Optimization methods for mechanical applications 38
3.2.1 Automatic sizing 38
3.2.2 Shape optimization 39
3.3 Topology optimization 40
3.3.1 Homogenization based methods 41
3.3.2 Level set method 48
3.4 Shape optimization of electrostatic microsystem 50
3.4.1 Pull-in voltage optimization 50
3.5 Topology optimization of microsystems 53
3.5.1 Electrothermal actuator optimization 53
3.5.2 Electrostatic actuator topology optimization 55
3.6 Conclusion 65

4 Simplified topology optimization problem for pull-in voltage maximization 67
4.1 Introduction 67
4.2 Simplified optimization problem 68
4.2.1 Optimization problem simplification 68
4.2.2 Sensitivity analysis 69
4.2.3 Optimization procedure 71
4.3 Numerical applications 71
4.3.1 Clamped-clamped microbeam suspension optimization 73
4.3.2 Microbeam suspension optimization with off-center electrode . . 75
4.3.3 Comparison with linear compliance optimization 76
4.4 Manufacturing constraint 81
4.4.1 Constraint formulation 82
4.4.2 Numerical applications 85
4.5 Conclusion 88

5 Eigenproblem formulation for the simplified optimization problem 91

5.1 Introduction . . . 91

5.2 Eigenproblem formulation . . . 93

5.2.1 One-point formulation . . . 93

5.2.2 Two-point formulation . . . 98

5.2.3 Condensation of the tangent stiffness matrix . . . 101

5.3 Double actuator model study . . . 105

5.4 Sensitivity analysis . . . 108

5.4.1 Eigenvalue sensitivity analysis . . . 108

5.4.2 Condensed stiffness matrix sensitivity . . . 109

5.4.3 Computation of $\frac{\partial \mathbf{K}_{xx}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_j}$ . . . 110

5.5 Numerical applications . . . 112

5.5.1 Optimization of a double actuator . . . 112

5.5.2 Pull-in voltage optimization of a microbridge . . . 115

5.5.3 Eigenproblem formulation for pull-in voltage optimization . . . 122

5.6 Optimization with repeated eigenvalues . . . 125

5.6.1 General framework . . . 125

5.6.2 Optimization methods for repeated eigenvalues . . . 127

5.6.3 Implementation of the optimization procedure . . . 131

5.6.4 Numerical application to the microbridge problem . . . 133

5.7 Sensitivity validation . . . 135

5.7.1 Comparison with pull-in voltage sensitivities . . . 135

5.7.2 Modification of the optimization procedure . . . 139

5.7.3 Summary . . . 141

5.8 Conclusion . . . 141

6 Generalization of pull-in voltage maximization 143

6.1 Introduction . . . 143

6.2 Generalized optimization problem . . . 144

6.2.1 Design problem description . . . 144

6.2.2 Electrostatic modeling modifications . . . 145

6.2.3 Material model choice . . . 150

6.3 Electrostatic forces computation . . . 155

6.3.1 Existence of local instability modes . . . 155
6.3.2 Origin of the local modes . . . 158
6.3.3 Solutions to prevent local modes . . . 161
6.4 Numerical application . . . 168
6.4.1 Pull-in voltage optimization . . . 169
6.4.2 Comparison with linear compliance optimization . . . 170
6.5 Conclusion . . . 172

7 Electrostatic actuator optimization 175
7.1 Introduction . . . 175
7.2 Electrostatic actuator design procedure . . . 176
7.2.1 Optimization problem . . . 176
7.2.2 Sensitivity analysis . . . 177
7.2.3 Optimization procedure implementation . . . 178
7.3 Numerical application . . . 178
7.3.1 Optimization problem . . . 178
7.3.2 Optimization for low input voltage . . . 180
7.3.3 Optimization for high input voltage . . . 183
7.3.4 Physical interpretation . . . 184
7.3.5 Summary . . . 186
7.4 Electrostatic actuator design including pull-in voltage constraint . . . 186
7.4.1 Optimization problem . . . 186
7.4.2 Modifications of the optimization procedure . . . 187
7.4.3 Numerical application . . . 189
7.5 Influence of large displacement modeling . . . 190
7.5.1 Mechanically nonlinear post-analysis of previous designs . . . 191
7.5.2 Actuator optimization including geometric nonlinearity . . . 192
7.6 Conclusion . . . 193

8 Conclusion 195
8.1 Summary of the work . . . 195
8.2 Achievements of the thesis . . . 197
8.3 Future works . . . 198
8.3.1 Extension to three-dimensional problems . . . 198

8.3.2 Suspension design for electrostatic oscillators 198
8.3.3 Introduction of additional design constraints 198
8.3.4 Reliability-based optimization 199
8.3.5 Larger gaps thanks to the pull-in voltage constraint 199
8.3.6 Generalization of the stability eigenproblem approach 200

A Semi-analytic derivative of the tangent stiffness matrix 203
A.1 Introduction 203
A.2 Electrostatic tangent stiffness 203
A.2.1 Derivative of $\mathbf{B}_\phi$ matrix 205
A.2.2 Derivative of the Jacobian matrix and of its determinant 205
A.3 Mechanical tangent stiffness $\mathbf{K}_{uu}$ 206
A.4 Electromechanical coupling tangent stiffness $\mathbf{K}_{\phi u}$ 208
A.5 Conclusion 208

Bibliography 209

List of symbols

Finite element formulation

$\mathbf{K}_C$	Condensed tangent stiffness matrix
$\mathbf{K}'_C$	Condensed tangent stiffness matrix derivative with respect to the load variable
$\mathbf{K}_t$	Tangent stiffness matrix
$\mathbf{K}_{\phi\phi}$	Electrostatic part of the tangent stiffness matrix
$\mathbf{K}_{uu}$	Mechanical part of the tangent stiffness matrix
$\mathbf{K}^0_{uu}$	Mechanical contribution to mechanical part of the tangent stiffness matrix
$\boldsymbol{\phi}$	Electric potential vector
$\mathbf{f}_m$	Mechanical forces vector
$\mathbf{f}_{es}$	Electrostatic forces vector
$\mathbf{f}_{ext}$	Generalized external forces vector
$\mathbf{f}_{int}$	Generalized internal forces vector
$\mathbf{q}$	Generalized displacements vector
$\mathbf{Q}_{es}$	Electric charges vector
$\mathbf{r}$	Residual forces vector
$\mathbf{u}$	Mechanical displacements vector

Material properties

ε	Dielectric permitivitty
ε_0	Void dielectric permitivitty

ρ_s	Design material density
E	Young modulus
E_s	Design material Young modulus
E_{min}	Young modulus of the soft material used to represent void in topology optimization

Miscellaneous

V_{eig}	Voltage at which the stability eigenproblem is formulated
$V_{pi,pred,k}$	Predicted pull-in voltage associated with k^{th} eigenvalue of the stability eigenproblem

Optimization

$\overline{v}$	Structure volume upper bound
p_ε	Permittivity penalty parameter (SIMP)
p_E	Young modulus penalty parameter (SIMP)
v	Structure volume

Chapter 1

Introduction

1.1 General context of the thesis

Nowadays, miniaturization is one of the keys to technology evolution. For about twenty years, industry is able to manufacture mechanical sensors and actuators smaller than one square millimeter. These devices are called MEMS (Micro ElectroMechanical Systems) because they combine electrical and mechanical components that can be as small as one micron. Some of the most famous applications of MEMS devices include: accelerometers or gyroscopes that can be found in almost all smart phones today or are used for ESP and ABS systems in vehicles, Digital Mirror Device chips for projection displays and pressure sensors for car tires. Size reduction enhances possibilities and allows lower costs, lower power consumption and better reliability. However, several challenges come with this downsizing.

At first we can think of manufacturing difficulties. To reach sub-millimetric sizes, mechanical parts of the devices were initially built using manufacturing processes adapted from electronic industry like lithography techniques [68]. Since that time, microfabrication methods have evolved and specific manufacturing methods have been developed such as LIGA or micromolding [68]. As these processes are very different from the macroscopic ones, they lead to unusual manufacturing constraints. Nevertheless, a great advantage of MEMS manufacturing techniques is that they generally comply very well with batch manufacturing in order to reduce the devices price.

Secondly, with the downscaling of the mechanical structure, physical effects that are often neglected at macroscopic scale become significant in MEMS and vice versa. For instance, magnetic, electrostatic, and thermomechanical forces may be larger than gravity forces. This can be seen as an advantage, since these forces can be used in order to create deformation or vibration of a part of the microsystem structure. However, it makes the modeling of microsystems more difficult, as it leads to strong couplings between several physical fields.

Consequently, the tools classically used to design at macroscopic scale cannot be applied

directly for MEMS design. The modeling methods and the optimization techniques must be adapted in order to take microscale specificities into account. In the domain of modeling software, different types of coupling strategies have been developed in order to model the interactions between several physical fields. In parallel, several research studies have been considering the development of optimization procedures dedicated to MEMS design [1, 56, 99, 129]. Classical optimization methods have been intended for problems including one single physical field. They have to be adapted to take into account the strong interactions between the multiple physical fields. New design criteria must also be developed to include MEMS distinctive behaviors. In addition, the unusual manufacturing constraints of microfabrication processes should be introduced into the optimization problem.

1.2 Objectives of the thesis

The present thesis is dedicated to the development of an optimization procedure for electrostatic microactuators. In these microdevices, an input voltage is applied to the device to create electrostatic forces which, in turn, generate mechanical deformations. Consequently, there is a strong coupling between mechanical and electrostatic fields as a modification of the electrostatic forces involves a deformation of the structure and vice versa. As a result of the coupling, the behavior of electrostatic microactuators is nonlinear. This thesis resorts to existing finite element formulation available in *Oofelie* software [76] to treat the strongly coupled modeling problem.

Moreover, the nonlinearity of the electrostatic actuators leads to an unstable behavior: the *pull-in effect*. The pull-in effect is analogous to buckling behavior as it corresponds to a limit point on the equilibrium curve. This phenomenon appears if a voltage greater than the so called *pull-in voltage* is applied to the device. Beyond this critical voltage, the actuator collapses and its electrodes stick together, potentially resulting in a permanent damage of the device. Clearly, the pull-in voltage is an important property of electrostatic actuators, which has to be considered during design. Therefore, including the pull-in voltage into the developed optimization procedure is a central objective of this thesis.

The optimization method selected for the present thesis is topology optimization based on homogenization approach. This technique offers several advantages over other optimization methods. Firstly, it is more general than sizing and shape optimization as it provides larger design freedom to the optimization. Secondly, its practical implementation is less complex than the level set methods. Moreover, applications of topology optimization to MEMS systems design are available in literature and offer promising results [56, 86, 99, 129].

The objective to establish a topology optimization procedure for electrostatic actuators able to control pull-in voltage has been divided in several sub-objectives:

- First of all, we need to be able to evaluate the pull-in voltage of a device and to develop a sensitivity analysis procedure for pull-in voltage. As sensitivity analysis is performed at each optimization iteration, the developed tool has to be as efficient as possible.

- Secondly, the material model classically used in topology optimization has to be extended in order to include electrostatic material properties. The choice of the material model is critical. Since the electrostatic forces are design dependent loads, an unsuitable material model can lead to local instability modes and to a failure of the optimization process.

- Additionally, the possibility to include manufacturing constraints in the optimization problem has to be investigated. We will consider the implementation of a constraint related to material deposition manufacturing processes.

- Once, the pull-in design function is developed and tested, the initial objective can be achieved by inserting the pull-in function as a constraint into a classical actuator design problem (e.g. maximizing the output displacement for a given input voltage).

All modeling and sensitivity analysis procedures developed in this thesis have been implemented into the C++ software *Oofelie* while *Conlin* optimizer [41] has been used to solve the optimization problem.

1.3 Layout of the thesis

After this introductory chapter, the two following chapters are mainly focused on the review of literature. This review covers the topics common to several or all of the chapters of the thesis. Moreover, for topics specific to one chapter, the literature review is proposed in the corresponding chapter. Next, Chapters 4 to 7 present the developments achieved along this work. Each of these chapters includes numerical applications in order to illustrate and also to justify the developments.

The outline of the thesis is presented in Figure 1.1. In Chapter 2, different practical applications of electrostatic actuation in MEMS are firstly presented. Then, the behavior of electrostatically actuated microsystems is studied and pull-in effect is explained on the basis of a simple model. Next, the existing modeling methods suited to microelectromechanical coupling simulation are reviewed. More specifically, the finite element framework used in the present thesis is summarized. Chapter 2 ends with the development of a path following algorithm able to compute the pull-in point. This procedure is the basis of the sensitivity analysis used in the pull-in voltage optimization.

The state of the art of optimization applied to microsystems is detailed in Chapter 3. At first, sizing, shape and topology optimization techniques usually used for single physic problem are described. Then, some of the existing extensions of optimization to

multiphysic microsystems are presented. Special attention is given to existing studies considering electrostatic actuation.

Using the essential modeling tools presented in Chapter 2 and the topology optimization method described in Chapter 3, we can start developing the pull-in voltage design function. For the development and the testing of the new design function, a basic optimization problem consisting in maximizing the pull-in voltage is considered. Moreover, in order to solve the difficulties one by one, we first make the assumption that the optimization and the electrostatic domains are separated resulting in a purely mechanical optimization domain.

Starting from this simplified optimization problem, a first approach to maximize pull-in voltage is presented in Chapter 4. In this approach we optimize the actual value of pull-in voltage as it can be computed by using the path following algorithm. The efficiency of the method is illustrated on numerical applications. Moreover, in the framework of the assumed separation between optimization and electrostatic domains, a manufacturing constraint is also proposed. This constraint aims at avoiding the creation of closed cavities in order to make the design easier to manufacture using a material layer deposition process.

Alternatively, instead of following the equilibrium curve up to the critical point, an approximation of the pull-in voltage can be determined by formulating a linear stability eigenproblem at some stable point of the equilibrium path. The development of this approximated approach is described in Chapter 5. Moreover, it is also possible to formulate an optimization problem on the basis of the resulting approximated pull-in voltage. The main advantage of this approach is that it avoids optimization oscillations that may happen in some applications if the actual value of pull-in voltage is optimized.

The promising results from Chapters 4 and 5 lead us to consider the generalization of the pull-in maximization problem in Chapter 6. The generalized problem broadens the design freedom by removing the assumption of separation of the optimization and electrostatic domains. Hence, the optimization domain becomes electromechanical and the location of electrostatic forces becomes design dependent. The generalization first requires the development of a new material model including both electrostatic and mechanical material properties. During the development it comes out that the approximated formulation proposed in Chapter 5 is unsuited to the generalization of the optimization problem. Therefore, only the approach from Chapter 4 is retained. Additionally, difficulties related to artificial local instability modes have to be investigated and circumvented. Chapter 6 finalizes the development of the pull-in voltage design function.

Next, Chapter 7 proposes the development of an actuator design procedure for electrostatic microsystems. At first, a classical actuator design formulation is considered. A numerical application shows that the optimization procedure may fail because of the pull-in effect. This situation can be avoided by including the developed pull-in voltage design function as a constraint into the actuator optimization problem. The new design

constraint enables the optimization problem to achieve the convergence without trouble. To conclude this chapter, the influence of large displacement modeling is investigated.

Finally, the conclusion of the thesis and the perspectives for future work are presented in Chapter 8.

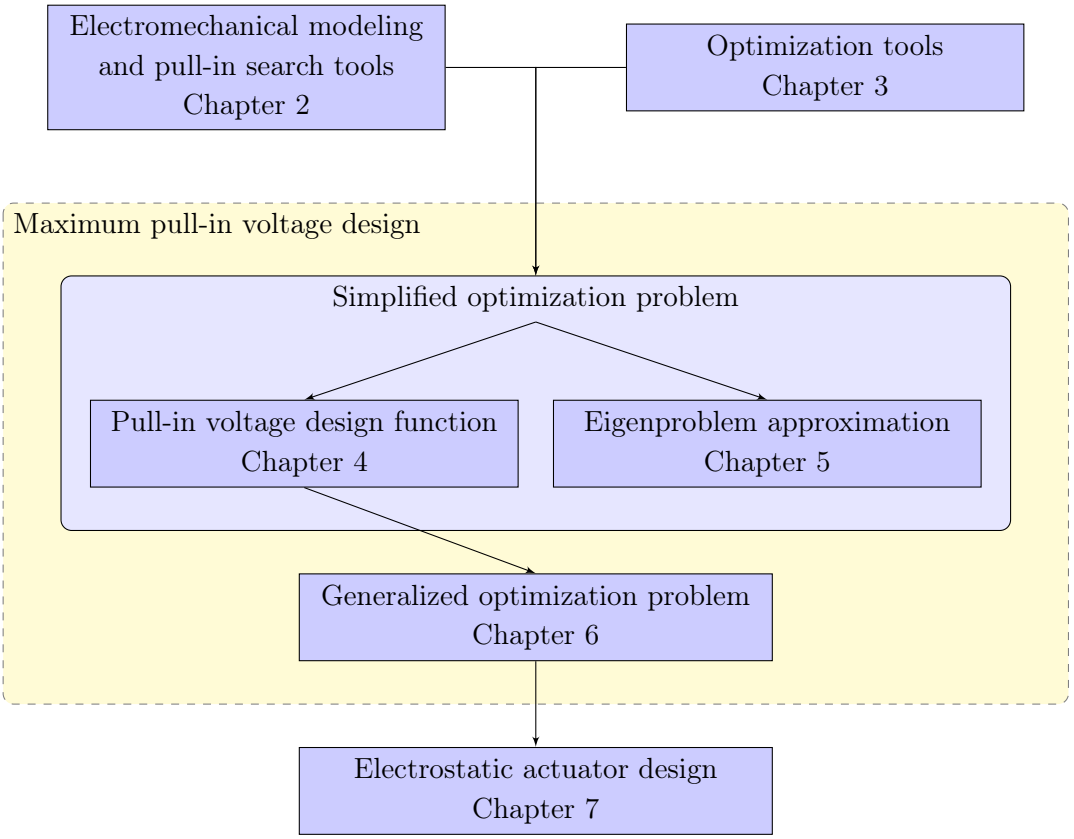


Figure 1.1: Layout of the thesis.

Chapter 2

Electromechanical modeling

2.1 Introduction

Electromechanical microsystems integrate mechanical components to electronic circuits to enhance sensing capabilities or to create microactuators. With the introduction of these mechanical parts comes the necessity to generate mechanical forces at microscale. However, because of the scale reduction, dominant forces in microsystems are not the same as at macroscopic scale. As described by Senturia [94], thermal expansion, electrostatic or piezoelectric forces have to be considered in microsystems and can be used as actuation forces. Moreover, it is not always possible to model separately the different physical phenomena that are involved in MEMS as there may be a strong interaction between them. Therefore dedicated tools have to be developed in order to model accurately the coupling between the physical fields.

In the scope of this work, devices using electrostatic forces are studied. The actuators used to generate electrostatic forces are in fact similar to capacitors. Basically, they are composed of two electrodes between which a voltage difference is applied. The generated electrostatic forces tend to bring the two electrodes closer. The displacement created by the electrostatic force can then be used in different ways as explained later.

This chapter is dedicated to the review of the behavior of electrostatically actuated MEMS and to a description of the methods used to model this behavior. As mentioned in the introduction chapter, these devices exhibit a strong coupling between electrostatic and mechanical physical fields. A consequence of this coupling is that the mechanical response to a voltage input is highly nonlinear and an unstable behavior is obtained if the input voltage exceeds a threshold called *pull-in voltage*. This instability, named pull-in effect, is very similar to buckling commonly known in mechanics. Indeed, alike pull-in, buckling appears when the mechanical load applied to a structure becomes greater than a limit value and results in a collapse (unstable behavior) of the structure. The description of pull-in effect is the topic of the third section of this chapter.

Because of their complex behavior, the modeling of electrostatically actuated microde-

vices requires specific techniques. In the scope of this thesis, two main tools have been used. At first, the physical coupling between electrostatic and mechanical effects is taken into account by using a monolithic finite element formulation developed by Rochus [91]. The review of the finite element formulation is proposed in the fourth section of this chapter. Secondly, to handle the nonlinear and potentially unstable behavior of the model, a path following procedure has been developed. Based on the normal flow algorithm proposed by Ragon [85] the path following procedure is presented in the last section of this chapter.

2.2 Electrostatic microsystems

As already stated in the introduction section, several physical phenomena can be used to generate mechanical forces in MEMS. Piezoelectric, electromagnetic, electrostatic and thermal expansion forces are common. A review and a comparison of actuation principles used in MEMS are proposed by Bell *et al.* [13]. This study compares maximum output displacement, maximum output force and frequency range provided by each type of actuator. It comes out that there is no ultimate actuator, which surpasses all others but that every actuator possesses strong and weak points. For instance, electrostatic combdrive actuators provide larger frequency range than thermal expansion actuators but smaller output force. However, with respect to other actuation principles, electrostatic actuation has the advantage to be easier to implement in practice. Indeed, electrostatic actuation is compatible with cheap and well-tested MEMS manufacturing process as CMOS [108]. Moreover, it is characterized by a low power consumption. For these reasons, electrostatic actuation is very popular in MEMS devices. This section presents a few examples among others of microsystems in which electrostatic actuation is involved.

2.2.1 RF switches

One of the most common types of electrostatically actuated microsystems is radio frequency (RF) switches. These devices are part of a larger group of microsystems named RF MEMS that regroup all MEMS dedicated to radio frequency integrated circuits as switches, filters or variable capacitors for example. The domain of RF MEMS is very large as shown by the comprehensive description of RF MEMS done by Varadan in his book [114]. We will focus here on devices that are based on electrostatic actuation like RF switches.

Radio frequency switches are mainly seen in telecommunication devices for signal routing or to adjust gain of amplifiers. This application covers a very wide frequency range starting from 1 MHz (AM band) and ending at 100 GHz (W band) so that different types of switch are used according to the actual working frequency. RF MEMS switches have two competitors, the first ones are the macroscale electromechanical switches and the second ones are the solid state switches. Because of their slow switching speed and low

resonance frequency, macroscale electromechanical switches (as electromagnetic relays) can only be used in low speed applications. Classical solid state switches have a very short response time and can be used at higher switching speed. However they introduce too much loss and do not provide sufficient isolation[1] when the signal frequency is higher than 1 GHz. RF MEMS possess some advantages and disadvantages over the two other switching techniques. Thanks to scale reduction, their switching speed is higher than their macroscale counterpart and can be of the magnitude of one microsecond [125]. However they are still slower than solid state switches where it can be as low as $3 \cdot 10^{-9}$ s (see [114]). Nevertheless RF MEMS provide lower insertion loss[2] and better isolation than solid state switch specifically when the signal frequency is higher than 1 GHz. Another advantage of RF MEMS switches is the very low amount of energy they consume.

A micrograph of a simple RF MEMS switch is presented in Figure 2.1. The principle of this device is very simple. When no voltage is applied to the gate, a small gap between the beam and the drain prevents the signal from passing from the source to the drain. If a voltage is applied to the gate, electrostatic force bends the beam down and the beam comes in contact with the drain to close the circuit. If they are very basic, these devices have one major drawback. Since there is a direct contact between the beam and the drain, an electric arc can happen when opening the switch.

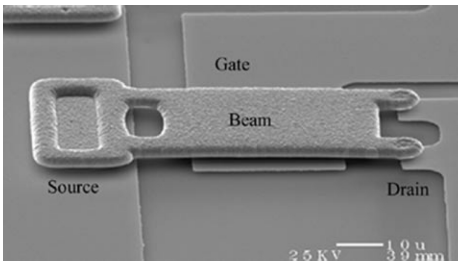


Figure 2.1: Metal contacting switch micrograph [125].

The electric arc generated when opening the switch may be harmful for the microdevice and reduces its reliability. Nevertheless, provided that the signal frequency is high enough, it is possible to avoid this problem by choosing another design. The microswitch presented in Figure 2.2 is different from the previous one as it integrates a layer of dielectric material on top of the drain electrode. This shunt switch design was originally proposed by Goldsmith at Texas Instrument [126]. The insulating layer acts as a spacer between the signal path and the drain (the metallic membrane) when the mobile electrode is down. This device works thanks to the capacitance variation between up-state and down-state. In down-state, the shorter distance between the signal path and the membrane leads to a higher capacitance than in up-state. The capacitance ratio can reach 100 depending on the geometry and material chosen. Provided that the input signal frequency is high enough (typically greater than 10 GHz), the signal is deviated

[1] Signal attenuation through the switch in blocking state.
[2] Signal attenuation through the switch in passing state.

from the 'signal path' to the 'ground' when the beam is down (high capacitance) while it is allowed to go through the switch when the beam is up. As there is no direct contact between the mobile electrode and the drain, electric arc appearance is avoided.

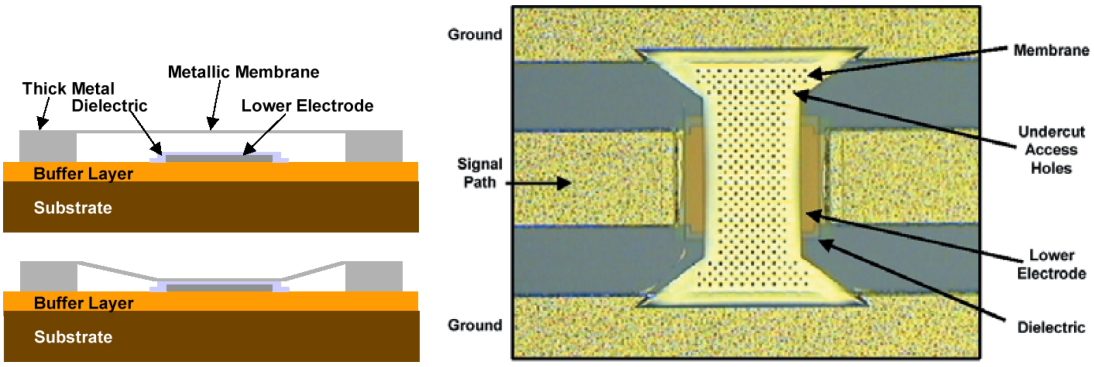


Figure 2.2: Capacitive coupling switch sketch and micrograph [125].

2.2.2 Microelectromechanical resonator

Like MEMS switches, microelectromechanical resonators are also introduced in radio frequency electronic circuits. Most of the time, they are used as band-pass filters which are made of several resonators coupled together. An example of such a filter coming from Ref. [115] is presented in Figure 2.3. The geometry of the system is more complex

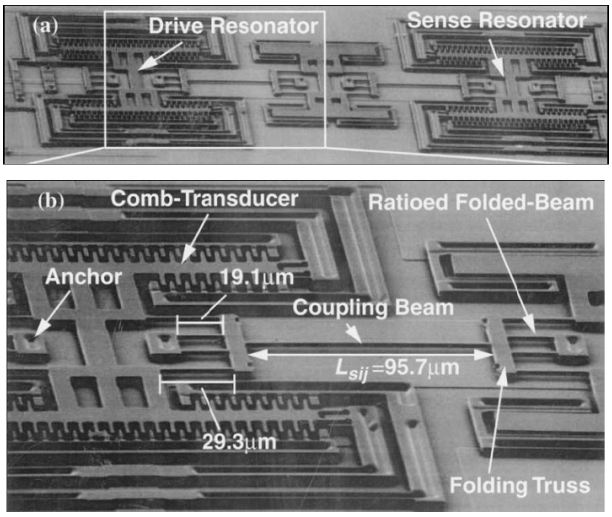
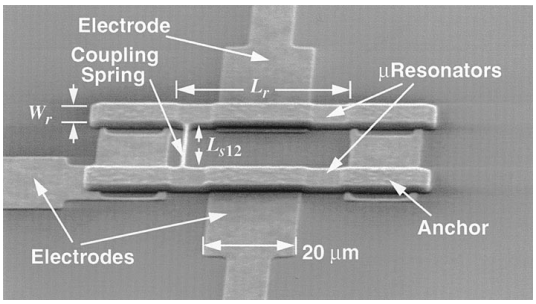


Figure 2.3: Micrographs of a microelectromechanical filter.

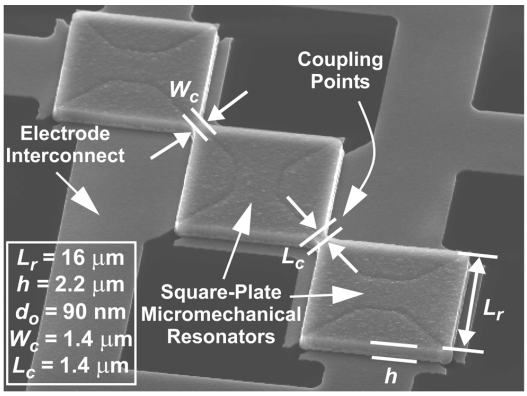
than the previous one. The filter is composed of three oscillators serially coupled using beams (coupling beams). The left one (drive resonator) is excited by electrostatic forces generated on the combdrive electrodes by the input signal. The central resonator has no combdrive electrodes but linear electrodes are available to allow tuning of its resonance

frequency. On the right, the combdrives attached to the sense resonator convert its movement into the output signal. These combdrives act as variable capacitors when the sense resonator moves. The output signal is generated by the combination of the sensing resonator displacement and of a bias voltage applied to the moving part. As the central resonator, the two other ones also possess linear electrodes that can be used to adjust their individual resonance frequency. These additional electrodes are very useful to correct imperfections due to fabrication tolerances and other parasitic interferences.

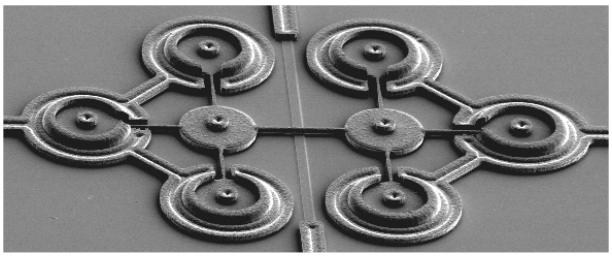
These devices also possess solid state and macroscopic electromechanical equivalents. However, their interest lies in the high quality factor they can achieve, their good temperature stability and aging properties [125]. Different designs exist depending on the frequency band targeted. The device presented in Figure 2.3 is designed for a central frequency of 300 kHz a band-pass width of 510 Hz and reaches a quality factor (Q factor) of 590. To reach higher central frequencies, beam resonators, square plate resonators and disk resonators have been developed. Using two mechanically coupled microbeams, Bannon (see [9] and Figure 2.4(a)) has been able to design a RF filter with a central frequency of 8 MHz and a quality factor reaching 435. Later, using corner-coupled square plate resonators, Demirci (see [31] and Figure 2.4(b)) created an RF MEMS filter characterized by a 70 MHz central frequency and a Q factor greater than 9000. Finally, a central frequency up to 200 MHz has been achieved using disk resonator (excited radially) by Li (see [58] and Figure 2.4(c)).



(a) RF filter by Bannon [9].



(b) RF filter by Demirci [31].



(c) RF filter by Li [58].

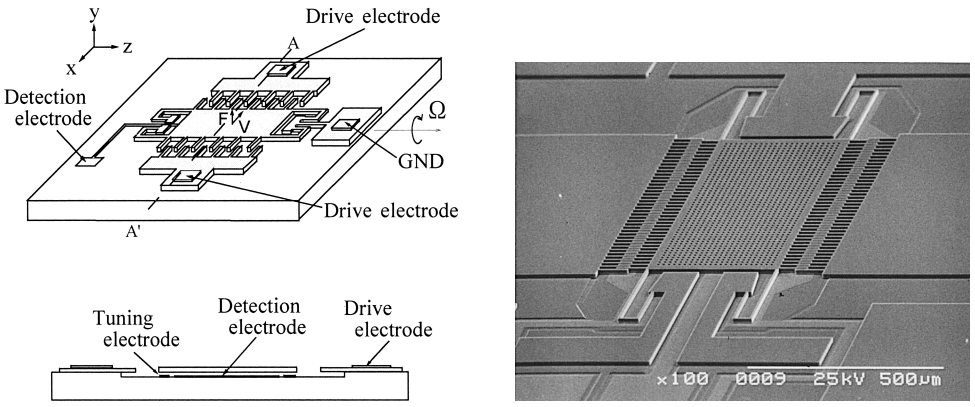
Figure 2.4: RF Filters designs.

2.2.3 Microelectromechanical gyroscopes

Gyroscopes are sensors used to measure the rotation rate of an object. This is another field where MEMS are taking more and more importance so that comprehensive reviews have been proposed (see for instance [2, 127]). The development of microgyroscopes has started more than two decades ago and microdevices able to measure rotation rate around one or even two axes are now commercially available.

Most of MEMS gyroscopes use the same physical principle involving Coriolis effect to detect rotation. A mass suspended by springs is vibrating linearly or angularly using electrostatic or electromagnetic actuation. As a consequence, if the device is rotated around one of its sensing axis, a Coriolis force appears on the oscillating mass and modifies its trajectory with respect to the device reference axes. The trajectory modification of the oscillator can be detected electrostatically (capacitance measure) or sometimes by using piezoresistors.

In his book [2], Acar describes several microgyroscope designs. As an example, Figure 2.5 presents some sketches and a micrograph of a simple MEMS gyroscope developed by Mochida *et al.* [71]. As shown by the sketches, the mobile mass is excited in x direction using the combdrive electrodes (Drive electrodes). A rotation of the device around z axis results then in an out-of-plane movement of the mobile mass. This rocking movement can be detected because of the capacitance modification using the electrode placed beneath the proof mass as indicated by the cross-section sketch.



(a) Gyroscope sketches.

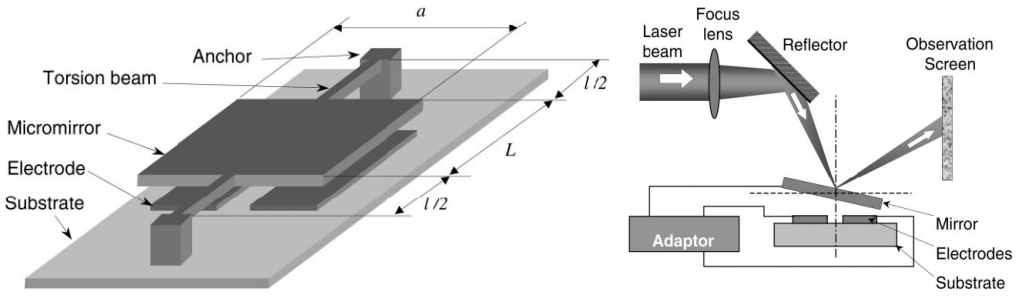
(b) Gyroscope micrograph.

Figure 2.5: Sketches and micrograph of a MEMS gyroscope [71].

2.2.4 Electrostatically actuated micromirrors

Optics is another domain where MEMS bring improvements and new possibilities. MOEMS (Micro OptoElectroMechanical Systems) is the name of this class of MEMS dedicated to optical applications. These microdevices can incorporate micromirrors, microlenses and optical wave guides. This section is focused on the presentation of

electrostatic micromirrors. The most famous application of micromirrors is the DMD (Digital Mirror Device) projection display by Texas Instrument [110]. The core of this display consists in a 2D array of tiltable micromirrors. Each mirror corresponds to a pixel of the image and can be turned 'on' and 'off' by using electrostatic actuation. The micromirrors are similar to the one presented in Figure 2.6(a). The electrodes placed beneath the mirror allow tilting the mirror around its suspension axis deflecting light towards or off the screen.



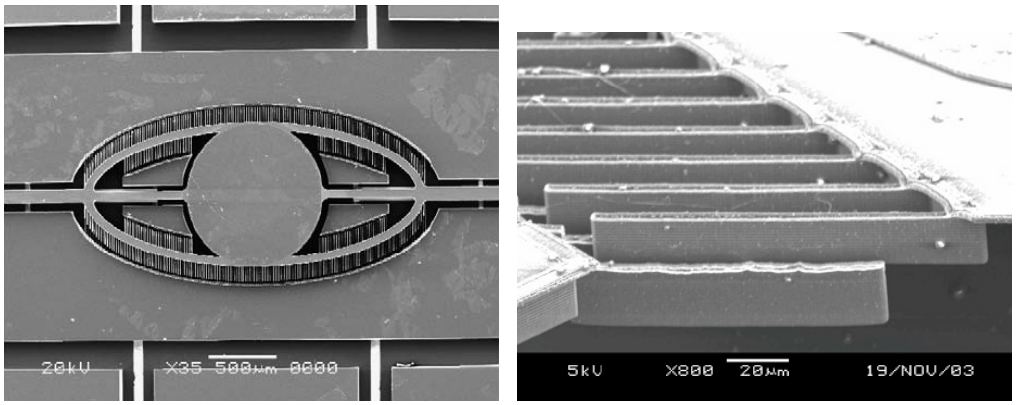
(a) Basic torsional micromirror sketch. (b) Functioning of a micromirror.

Figure 2.6: Tortional micromirror principle [130].

Basically, micromirrors used in DMD only need to achieve two discrete positions corresponding to 'on' and 'off' states. However, another concept for projection display is the scanning display [51]. In scanning displays, rather than using an array of micromirrors to create a complete image at once, the image is created by a light beam that covers the complete area by a scanning process as illustrated in Figure 2.6(b). The main advantage of this technique is that only two mirrors are needed (one for each direction). Conversely to DMD these mirrors have to be able to cover continuously the complete range of tilting angle. Moreover, aside from display applications, scanning micromirrors are also used in bar code readers, confocal microscopes and laser printers to replace the classic galvanometric mirror scanners.

A scanning micromirror design developed by Ko *et al.* [51] is presented in Figure 2.7(a). In this design, the planar electrodes are replaced by vertical combdrives to increase actuation force. Conversely to the coplanar combdrives used in microresonator design in Figure 2.3, the two combs of vertical combdrives are here shifted vertically as illustrated in Figure 2.7(b). This vertical offset results in vertical electrostatic forces that generate a torsion moment and tilt the mirror. In Figure 2.7(a), the central circular part is the mirror itself. The mirror is supported by an oval frame upon which actuating combs are placed. With the micromirror they developed, Ko *et al.* [51] have been able to setup a laser scanning display. They use the micromirror for high frequency horizontal scanning while the vertical scan is performed by a galvanometric mirror.

More recently, a two-dimensional electrostatic microscanner design has been proposed by Chu and Hane [28]. In this system, the mirror is supported by a gimbal frame whose rotation axis is perpendicular to the mirror one. Consequently, the device can carry



(a) Eye-type scanning mirror by Ko *et al.* [51]. (b) Vertical combdrive by Wu *et al.* [122].

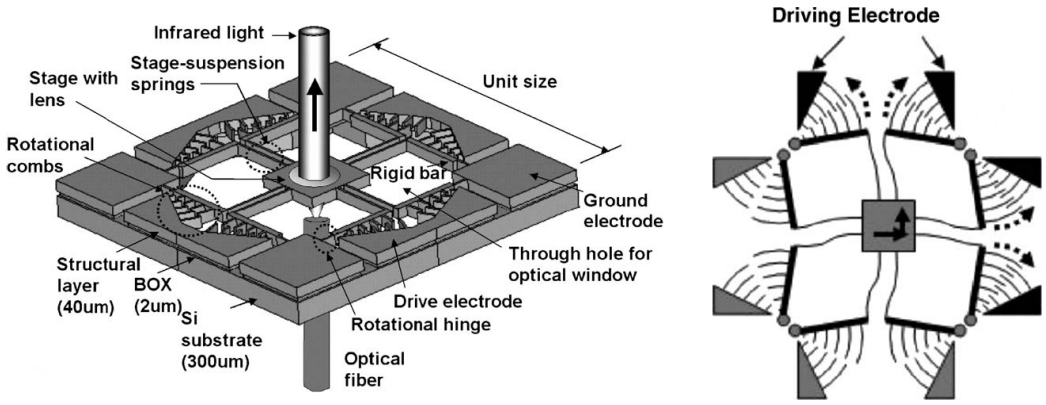
Figure 2.7: Example of a torsional micromirror design.

out both horizontal and vertical scanning and avoids the needs for a second scanning mirror.

2.2.5 Microlens optical scanner

Aside from mirrors, lenses are also included in MEMS. An application of microlenses is optical cross connection (OXC) into optical switches. Basically, the lenses are used to steer light between two optical fibers arrays in order to send light from each input fiber to the right output fiber. Of course, this task could be achieved by micromirrors as those presented in last section. Nevertheless, the availability of microlens scanners allows sparing one component since the light coming from input optical fibers needs in any case to be collimated by a lens. Moreover, microlens scanner can easily compensate for deviation of the fiber position and avoid costly accurate positioning of a passive lens. Concerning performances, optomechanical switches present longer switching time (of the order of one millisecond) compared to their electronic counterparts. However conversely to electronic switching, they do not introduce any bottleneck by requiring conversion of the light signal into an electronic signal and the bandwidth of the optical network is preserved (see Ref. [107]).

Kwon *et al.* [54] present a design of microlens scanner as the association of a microlens and a MEMS *XY-stage* (see Figure 2.8(a)). The XY-stage role is to perform biaxial translation of the microlens in a plane parallel to the substrate to achieve adequate positioning of the lens with respect to the light ray (perpendicular to the substrate). In the device proposed by Kwon *et al.* electrostatic actuation is used to achieve this goal. The XY-stage is equipped with four pairs of electrostatic actuators that allow translating the lens along both directions. This is illustrated in Figure 2.8(b) where a voltage is applied on black electrodes while gray electrodes are disabled to achieve biaxial displacement. The XY-stage developed by Kwon *et al.* is able to move the lens of more than 55 μm.



(a) 3D view of the optical scanner.

(b) Bidirectional actuation.

Figure 2.8: Two-dimensional optical scanner by Kwon *et al.* [54].

2.3 Insight of the behavior of the electrostatic actuator

As described in the previous section, electrostatic actuation is a convenient and efficient way to produce forces in microelectromechanical devices. However, the nonlinearity of the electrostatic force with respect to the gap separating electrodes results in phenomena such as the pull-in effect. Pull-in effect is related to an unstable behavior of the actuator when the voltage exceeds an upper limit value called the pull-in voltage. Physically speaking, this effect is very similar to nonlinear buckling occurring in mechanical structures or to divergence of aeroelastic wings.

To explain the behavior of electrostatic actuators, let's consider the simplified device represented in Figure 2.9. This capacitive system is made up of two parallels and rigid plates with the upper plate suspended by a linear spring of stiffness k and the lower plate fixed.

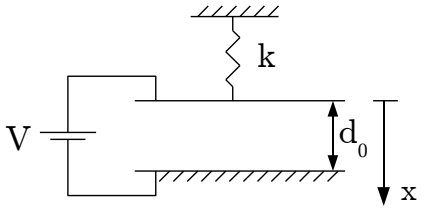


Figure 2.9: Simplified electromechanical actuator.

If we assume the plates large enough so that side effects are negligible, the electrostatic force resulting from the application of a voltage difference V between the plates can be written [87],

$$f_{es} = \frac{\varepsilon_0}{2} \frac{AV^2}{(d_0 - x)^2}, \tag{2.1}$$

where ε_0 denotes the permittivity of the media separating the plates (e.g. vacuum or

air), A is the capacitor surface, d_0 the initial distance between the plates and x the displacement of the upper plate. Moreover, the restoring force from the spring is simply given by:

$$f_r = -kx.$$

After applying the Newton's second law and a few manipulation of the resulting equation, we can obtain the following equilibrium equation giving the voltage as a function of the displacement:

$$V = \sqrt{\frac{2kx\left(d_0 - x\right)^2}{\varepsilon_0 A}}. \tag{2.2}$$

Figure 2.10 plots the normalized equilibrium equation in the plane (x, V). We can see that this curve possesses a maximum in terms of voltage for $x = d_0/3$. This maximum corresponds to the pull-in voltage (V_{pi}) and no equilibrium position can be found for a greater voltage. Using Eq. (2.2), we can compute the value of pull-in voltage by substituting $x = d_0/3$:

$$V_{pi} = \sqrt{\frac{8kd_0^3}{27\varepsilon_0 A}}. \tag{2.3}$$

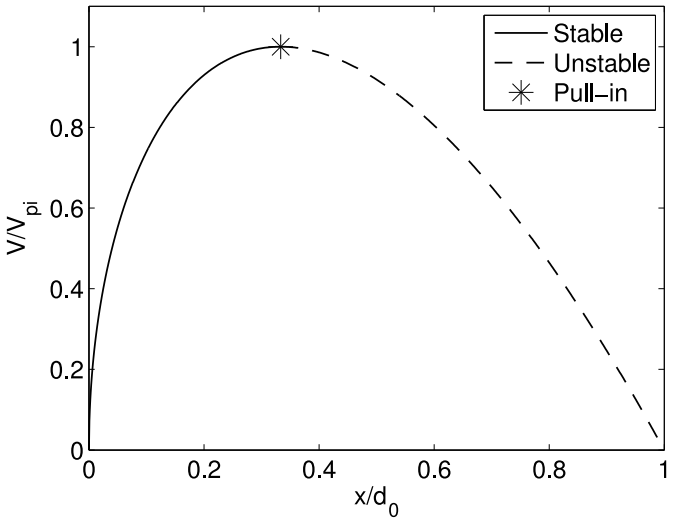


Figure 2.10: Normalized equilibrium curve of the simplified actuator.

By analogy with structural buckling, the pull-in voltage corresponds to a limit load. In practice, the application of an electric potential creates an electrostatic force, which tends to bring the electrodes closer. However, as the upper electrode goes down, the electrostatic force increases proportionally to the inverse of the squared gap (see equation (2.1)). Consequently, at pull-in point, the linear spring cannot balance anymore the raising of the electrostatic force and the mobile electrode collapses toward the fixed one.

In addition, the pull-in point divides the equilibrium curve in two parts: a stable one (solid line in Figure 2.10) and an unstable one (dashed line in Figure 2.10). This stability

inversion can be checked by computing the effective stiffness (i.e. tangent stiffness) along the equilibrium path (x_e, V_e). The effective stiffness is simply obtained by computing the derivative of the total force $f = f_{es} + f_r$ applied to the mobile electrode with respect to its displacement.

$$k_{eff} = -\frac{df}{dx_e} = k - \frac{\varepsilon_0 A V_e^2}{(d_0 - x_e)^3} = k - \frac{2kx_e}{d_0 - x_e}.$$

The influence of the applied voltage on the tangent stiffness is plotted in Figure 2.11. Starting from one (mechanical stiffness of the spring) when no voltage is applied, the normalized tangent stiffness decreases when the voltage grows. Eventually, the tangent stiffness vanishes when the pull-in voltage is applied and becomes negative if we proceed on the equilibrium curve. It results in an unstable equilibrium position at the pull-in point and for larger values of the displacement. This property can be used to detect the pull-in point.

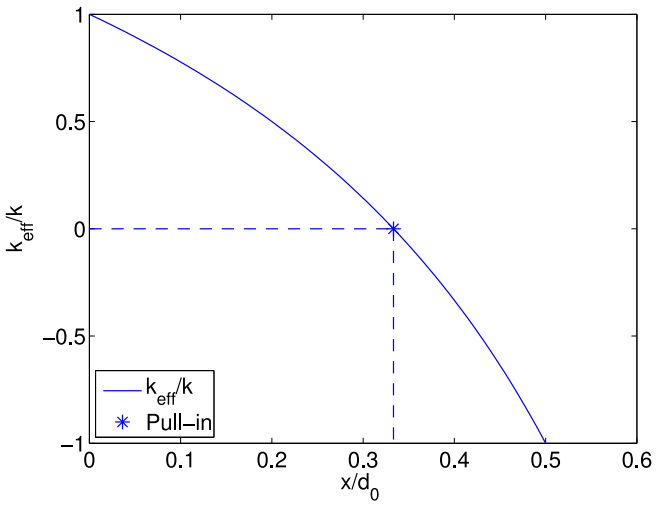


Figure 2.11: Normalized tangent stiffness against displacement on equilibrium curve.

Depending on the application, the pull-in effect may be desirable or undesirable. Indeed, we have seen in the last section that RF-switches rely on a collapse of the mobile electrode. Conversely, pull-in effect has to be avoided in some MOEMS, in RF-filters and in gyroscopes. In the later devices pull-in may result in contact of the combdrive electrodes but may also lead to a collapse of the proof mass on the substrate. Unnecessary contact is usually avoided in MEMS since firstly it can create a short circuit and may damage the microsystem and secondly even if a dielectric layer is placed to prevent short circuit, it can be impossible to separate the electrodes from each other after pull-in in wet environment (see Ref. [113]). Therefore, pull-in is an important phenomenon that has to be taken into account during MEMS design and this is the reason why this thesis proposes to introduce pull-in voltage criteria into topology optimization of MEMS.

2.4 Electromechanical finite element formulation

The finite element method is used in this thesis to model the behavior of electrostatic microdevices. The results provided by the finite element analysis will then be the basis of the electrostatic microdevice topology optimization. However, several finite element formulations are available in the literature and the one that suits the best the problem has to be selected. For multiphysic problems, finite element methods can be classified into two categories, *staggered* methods and *monolithic* formulations.

Staggered methods are the simplest to develop. Indeed these methods use sequential and separated computation for each physical field of the analysis domain as illustrated in Figure 2.12. In this way, it is possible to use different software codes for each physical field and take advantage of existing software tools. Depending on the coupling type, several iterations are needed to reach equilibrium between every physical field.

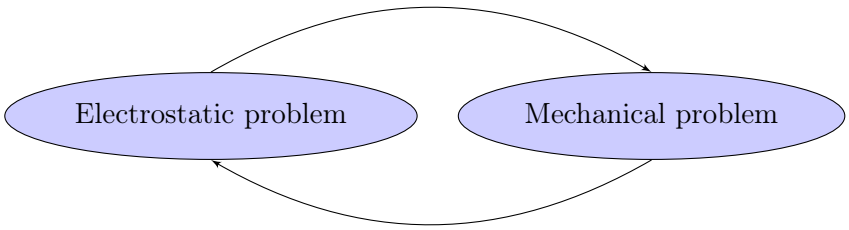


Figure 2.12: Staggered modeling procedure

However, staggered methods may lack convergence when the interactions between the physical domains increase. Therefore it is sometimes required to resort to the second class of methods: monolithic formulations. Monolithic formulations gather several physics into one single global problem so that the different physical fields are solved simultaneously. One disadvantage of monolithic approaches is that existing software tools cannot be used and a specific formulation has to be developed.

The choice of the modeling method to use for electromechanical microdevices is discussed by Rochus *et al.* [91]. This paper shows that when approaching pull-in, staggered methods lack convergence while monolithic methods remain reliable. Moreover, as it is described later, the pull-in voltage sensitivity analysis needed by the optimization process requires an accurate knowledge of the pull-in conditions. This means that we have to be able to reach exactly pull-in point which is not possible with a sufficient accuracy using staggered methods. Therefore, the electromechanical monolithic formulation developed by Rochus is used in this thesis.

2.4.1 Variational approach

In order to achieve an electromechanical monolithic formulation, Rochus *et al.* [91] start from the internal work by integrating over the domain the Gibbs energy density

G defined as follow:

$$G = \frac{1}{2}\mathsf{S}^T\mathsf{T} - \frac{1}{2}\mathbf{D}^T\mathbf{E}\,. \tag{2.4}$$

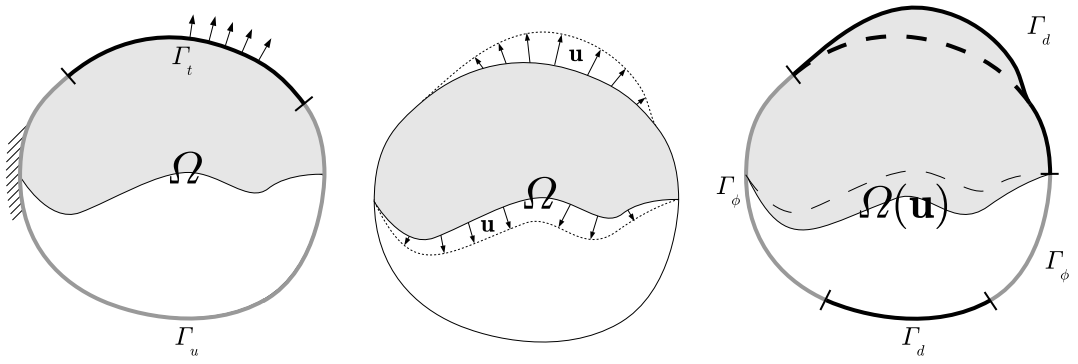
Each term of the Gibbs energy corresponds to a physical field of the problem. The first one takes into account the mechanical part of the energy with the product between the strain tensor S and the stress tensor T, while the second term represents the electrostatic energy contribution with $\mathbf{D}$ the electric displacement vector and $\mathbf{E}$ the electric field vector. These physical quantities are related by the constitutive laws where H denotes the elasticity tensor and $\boldsymbol{\varepsilon}$ the dielectric tensor.

$$\begin{cases} \mathsf{T} = \mathsf{H}\,\mathsf{S}, \\ \mathbf{D} = \boldsymbol{\varepsilon}\mathbf{E}. \end{cases}$$

The internal energy of the system is computed by integrating the Gibbs energy density from Eq. (2.4) over the whole electromechanical domain Ω.

$$W_{int} = \frac{1}{2}\int_{\Omega} \mathsf{S}^T\mathsf{T} - \mathbf{D}^T\mathbf{E}\, d\Omega = \underbrace{\frac{1}{2}\int_{\Omega} \mathsf{S}^T\mathsf{T}\, d\Omega}_{W_m} - \underbrace{\frac{1}{2}\int_{\Omega(\mathbf{u})} \mathbf{D}^T\mathbf{E}\, d\Omega}_{W_e}\,. \tag{2.5}$$

It is important to note here the difference between the mechanical energy W_m integration domain and the electrical energy W_e integration domain. Indeed, the mechanical energy is integrated over the reference domain Ω using a classical Lagrangian formulation. Conversely, the electrostatic energy depends on the mechanical displacements $\mathbf{u}$ and has to be integrated over the deformed domain $\Omega(\mathbf{u})$. These different integration domains are described in Figure 2.13 for a simple analysis domain containing two different materials. The mechanical energy integration domain is represented in Figure 2.13(a). It corresponds to the undeformed configuration. Conversely, the electrostatic energy integration domain has to take the mechanical deformation presented in Figure 2.13(b) into account. Therefore, the electric integration domain is shown in Figure 2.13(c).



(a) Mechanical domain. (b) Mechanical deformations. (c) Electrical domain.

Figure 2.13: Integration domains and boundary conditions descriptions.

The unknowns of the problem are the mechanical displacements $\mathbf{u}$ and the electric potential ϕ. These fields are discretized on each node of the finite element mesh. These unknowns are linked to the strains and to the electric potential by the compatibility relations:

$$\begin{cases} \mathsf{S}_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right), \\ \mathbf{E} = -\nabla\phi. \end{cases}$$

In addition, the boundary conditions of the problem are defined for each physical field (Fig. 2.13):

$$\begin{cases} \mathbf{u} = \overline{\mathbf{u}} & \text{over} \quad \Gamma_u, \\ \mathbf{t} = \overline{\mathbf{t}} & \text{over} \quad \Gamma_t, \end{cases} \qquad \begin{cases} \phi = \overline{\phi} & \text{over} \quad \Gamma_\phi, \\ d = \overline{d} & \text{over} \quad \Gamma_d, \end{cases}$$

where $\mathbf{t}$ denotes the surface tractions, d the electric surface charges while Γ_u, Γ_t, Γ_ϕ and Γ_d are the portions of the electromechanical domain boundary Γ where respectively $\mathbf{u}$, $\mathbf{t}$, ϕ and d have prescribed values. These four parts verify the following relations:

$$\begin{aligned} \Gamma_u \cup \Gamma_t = \Gamma, \quad \Gamma_\phi \cup \Gamma_d = \Gamma, \\ \Gamma_u \cap \Gamma_t = \emptyset, \quad \Gamma_\phi \cap \Gamma_d = \emptyset. \end{aligned}$$

The external energy can then be expressed as follow:

$$W_{ext} = \int_\Omega \mathbf{u}^T \overline{\mathbf{f}}\, d\Omega + \int_{\Gamma_t} \mathbf{u}^T \overline{\mathbf{t}}\, d\Gamma - \int_\Omega \phi\overline{\rho}\, d\Omega - \int_{\Gamma_d} \phi\overline{d}\, d\Gamma,$$

where $\overline{\mathbf{f}}$ denotes the imposed body forces and $\overline{\rho}$ the prescribed volume charge density over domain Ω. Let us notice that in the chosen formulation the nodal electric charges are analogous to the mechanical forces.

Next, the virtual work principle is applied by perturbing the state variables $\mathbf{u}$ and ϕ by kinematically compatible virtual displacements $\delta\mathbf{u}$ and $\delta\phi$ and by imposing that the resultant variations of the internal energy δW_{int} and the external energy δW_{ext} have to be equal. One of the specific feature of this procedure lies into the computation of the internal electrostatic energy W_e (see Eq. (2.5)). Indeed, since the integration domain of W_e depends on the displacements $\mathbf{u}$ it is also impacted by the virtual displacements $\delta\mathbf{u}$. This difficulty is circumvented by resorting to a variable substitution in order to express the integral of the perturbed electrostatic energy over the unperturbed domain. The computational details are available in Ref. [91].

2.4.2 Tangent stiffness matrix

The application of the variational approach to Gibbs energy leads to a linearized equilibrium equation of the strongly coupled electromechanical problem in Eq. (2.6). In this equation appear: the tangent stiffness matrix $\mathbf{K}_t$, the generalized displacements increments $\Delta\mathbf{q}$ and the corresponding generalized forces increment $\Delta\mathbf{g}$. Generalized

displacements are composed of both the nodal electric potentials $\boldsymbol{\phi}$ and the nodal mechanical displacements $\mathbf{u}$ while generalized forces are collecting the electric charges $\mathbf{Q}_{es}$ and the mechanical forces $\mathbf{f}_m$.

$$\underbrace{\begin{bmatrix} -\frac{\partial^2 W_e}{\partial \boldsymbol{\phi}^2} & -\frac{\partial^2 W_e}{\partial \boldsymbol{\phi} \partial \mathbf{u}} \\ -\frac{\partial^2 W_e}{\partial \mathbf{u} \partial \boldsymbol{\phi}} & \frac{\partial^2 W_m}{\partial \mathbf{u}^2} - \frac{\partial^2 W_e}{\partial \mathbf{u}^2} \end{bmatrix}}_{\mathbf{K}_t} \underbrace{\begin{bmatrix} \Delta\boldsymbol{\phi} \\ \Delta\mathbf{u} \end{bmatrix}}_{\Delta\mathbf{q}} = \begin{bmatrix} \mathbf{K}_{\phi\phi} & \mathbf{K}_{\phi u} \\ \mathbf{K}_{u\phi} & \mathbf{K}_{uu} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\phi} \\ \Delta\mathbf{u} \end{bmatrix} = \underbrace{\begin{bmatrix} \Delta\mathbf{Q}_{es} \\ \Delta\mathbf{f}_m \end{bmatrix}}_{\Delta\mathbf{g}}. \quad (2.6)$$

As suggested by Eq. (2.6), the tangent stiffness matrix can be split into four blocks. The first one, $\mathbf{K}_{uu}$, links the mechanical displacements to mechanical forces. According to the physical origin of the contributions to this block, we can write:

$$\mathbf{K}_{uu} = \mathbf{K}^0_{uu} + \mathbf{K}^*_{uu}.$$

Beyond the classical mechanical energy contribution $\mathbf{K}^0_{uu}$ the usual mechanical stiffness, $\mathbf{K}_{uu}$ includes also a contribution from electrical energy $\mathbf{K}^*_{uu}$. This later contribution comes from the dependency of the electrical energy on mechanical displacements. The influence of the electrostatic effects on the mechanical stiffness has been pointed out in Section 2.3. Indeed, while studying the simplified electromechanical actuator, we have seen that the linearized stiffness decreases when the input voltage is increased.

Secondly, the off-diagonal terms of the matrix generate a coupling between mechanical and electrical unknowns. The origin of these terms is also the dependency of the electrical energy on mechanical displacements. It is indeed clear that mechanical displacement results in a modification of the electric charge distribution and that conversely, a variation of the electric potential distribution modifies the electrostatic forces.

Finally, $\mathbf{K}_{\phi\phi}$ is simply equal to the pure electrostatic problem stiffness matrix. This comes from the fact that mechanical energy does not directly depend on the electric potential. Indeed, the displacements being fixed, a voltage modification leaves the mechanical energy unchanged (as far as piezoelectricity is ignored). Consequently, the only contribution to this last block comes from electrostatic energy.

2.4.3 Summary

The electromechanical coupling can be modeled using two different approaches: staggered or monolithic. The monolithic approach requires the development of specific finite element formulation but allows solving both physical fields at once. Moreover, within the monolithic finite element formulation, it is possible to write the linearized equilibrium equations. The related tangent stiffness matrix exhibits coupling terms linking the two physical fields. The monolithic nature of the formulation as well as the knowledge of the tangent stiffness matrix are two important assets as they enable us to use very efficient solvers like numerical continuation procedure. This association results in an accurate computation of the pull-in configuration of the studied electromechanical systems.

2.5 Normal flow algorithm

The optimization methods developed in this thesis require the ability to follow precisely the equilibrium curve of a microelectromechanical device and to locate limit points of this curve. However, this task cannot be achieved by the classical Newton-Raphson algorithm that is widely used in nonlinear mechanics. Indeed, Newton-Raphson method computes the response of the system for a given load level (i.e. the voltage in the present case). Therefore, theoretically, it would only be possible to compute the stable part of the equilibrium curve. However in practice, as explained by Rochus *et al.* [91], it is hardly possible to compute the stable part of the equilibrium curve since convergence is poor close to pull-in voltage. Hence pull-in conditions cannot be determined accurately.

Another option consists in resorting to path following algorithms. These methods are more sophisticated than Newton-Raphson procedure but allow following the equilibrium curve from rest position and enable passing singular points like pull-in point. Using this ability to compute the complete equilibrium curve (stable and unstable part) it is then possible to locate pull-in point accurately.

Path following methods differ from Newton-Raphson by the fact that the load parameter is considered as a variable of the problem and can be adjusted to restore equilibrium alike state variables of the system. With this extra variable the system owns $n+1$ variable for n equations (the equilibrium equations). Consequently an additional constraint (i.e. equation) is needed in order to close the system.

The various path following methods differ mainly in the nature of this supplementary constraint. Riks-Wemper method (see Refs. [85, 119]) constrains consecutive points to lie on a plane. Crisfield method (see Refs. [30, 59, 91]), also known as Arc-length method, is similar but the constraint plane is replaced by a sphere centered on last converged point. Normal flow algorithm (see Refs. [1, 85]) makes use of Moore-Penrose pseudoinverse to solve the $n \times (n+1)$ system of equations, which can also be interpreted as a constraint as explained in what follows.

However, it is difficult to determine the most efficient among available path following methods. The performance of the algorithms is problem-specific and depends on the quality of the implementation. Nevertheless, the three methods cited previously have been compared by Ragon *et al.* [85] on the basis of a nonlinear buckling benchmark. They conclude that normal flow algorithm is more efficient and more robust than the two other methods in buckling problems. Later, Abdalla *et al.* [1] have successfully applied normal flow algorithm to trace the equilibrium curve of electromechanical microdevices models. That is for these reasons that we select normal flow algorithm in the present thesis.

2.5.1 Homotopy methods

The normal flow algorithm is part of *homotopy methods* group also called *continuation methods*. Homotopy methods are numerical algorithms that have been developed to

solve systems of nonlinear equations when a good estimate of the solution is unknown a priori. The mathematical aspects as well as various implementations of homotopy methods are described in text books by Allgower [5, 6].

Originally, the problems targeted by homotopy methods are systems of N nonlinear equation in N variables that can be expressed in the general way:

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}. \tag{2.7}$$

$\mathbf{f}(\mathbf{x})$ can be considered as a mapping from $\mathbb{R}^N$ to $\mathbb{R}^N$. The standard procedure to solve this type of problem is to use a Newton-type algorithm. Starting from a point $\mathbf{x}_0 \in \mathbb{R}^N$ we can compute the sequence $\mathbf{x}_i$ in $\mathbb{R}^N$ such that:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - [\mathbf{Df}(\mathbf{x}_i)]^{-1} \mathbf{f}(\mathbf{x}_i),$$

where $[\mathbf{Df}(\mathbf{x}_i)]$ stands for the Jacobian matrix of $\mathbf{f}(\mathbf{x}_i)$. If $\overline{\mathbf{x}}$ is a solution of the system of equations and if $\mathbf{x}_0$ is sufficiently close to this solution, the sequence $\{\mathbf{x}_i\}$ finally converges to $\overline{\mathbf{x}}$. However, choosing $\mathbf{x}_0$ close to $\overline{\mathbf{x}}$ generally requires preliminary knowledge of the problem. Without this knowledge, Newton method often fails because poor starting points are chosen. One solution to overcome the problem is to define a homotopy function $\mathbf{h} : \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N$ such that:

$$\begin{cases} \mathbf{h}(\mathbf{x}, 0) = \mathbf{f}(\mathbf{x}), \\ \mathbf{h}(\mathbf{x}, 1) = \mathbf{g}(\mathbf{x}), \end{cases} \tag{2.8}$$

where the function $\mathbf{g} : \mathbb{R}^N \to \mathbb{R}^N$ has known zeros. Consequently, if we suppose that $\mathbf{x}_0$ is one of the zeros of $\mathbf{g}$, the point $(\mathbf{x}_0, 1)$ will be a zero of $\mathbf{h}$. The definition of the homotopy function in equation (2.8) leaves a lot of freedom on the actual mapping between the two functions. Amongst others, a commonly used homotopy is the convex homotopy:

$$\mathbf{h}(\mathbf{z}) = \mathbf{h}(\mathbf{x}, \lambda) = (1 - \lambda)\mathbf{f}(\mathbf{x}) + \lambda \mathbf{g}(\mathbf{x}),$$

where we introduce $\mathbf{z} = [\mathbf{x}, \lambda]$ a vector from $\mathbb{R}^{n+1}$ that will be used in the following because of its conciseness.

The idea of continuation method is to follow a curve $\mathbf{c}(s) : \mathbb{R} \to \mathbb{R}^N \times \mathbb{R}$ starting from $(\mathbf{x}_0, 1)$ such that $\mathbf{h}(\mathbf{c}(s)) = \mathbf{0}$ along the curve. Under conditions on smoothness of $\mathbf{h}$ and on existence of a solution to Eq. (2.7) as discussed in the book by Allgower [6], the curve $\mathbf{c}$ leads to a point $(\overline{\mathbf{x}}, 0)$ where $\overline{\mathbf{x}}$ is a zero point of $\mathbf{f}$.

Several methods have been developed to numerically follow the $\mathbf{c}(s)$ curve. In Ref. [6] Allgower classifies them into two groups: predictor-corrector methods and piecewise-linear methods. In predictor-corrector methods, the exact curve is approximately followed by a sequence of points. The gap between the points and the curve depends on the stopping criteria of the correction process. For piecewise-linear methods one follows exactly a piecewise linear curve that approximates the actual curve $\mathbf{c}(s)$.

2.5.2 Normal flow algorithm mathematical formulation

The normal flow algorithm belongs to the class of predictor-corrector methods. Figure 2.14 summarizes the steps of a predictor-corrector iteration where the solid line represents the curve $\mathbf{h}(\mathbf{z}) = \mathbf{0}$. Starting from a known point on the curve $\mathbf{z}_k$, one first tries to guess (or predict) the location of the next point $\mathbf{z}_{k+1}$ of the curve. The *prediction* $\tilde{\mathbf{z}}_{k+1,0}$ can be obtained tangentially or by using higher order information. The *prediction process* is controlled by the progression step h, which is a measure of the distance between points $\mathbf{z}_k$ and $\tilde{\mathbf{z}}_{k+1,0}$. The value of h is usually chosen by the user and can possibly be adapted by the algorithm to improve convergence properties. As the predicted point $\tilde{\mathbf{z}}_{k+1,0}$ does not usually lie on the curve, a *correction phase* is then applied to return on the curve by computing a secondary sequence of points $\{\tilde{\mathbf{z}}_{k+1,0}, \ldots, \tilde{\mathbf{z}}_{k+1,i}, \tilde{\mathbf{z}}_{k+1,i+1} \ldots\}$ that converges towards $\mathbf{z}_{k+1}$.

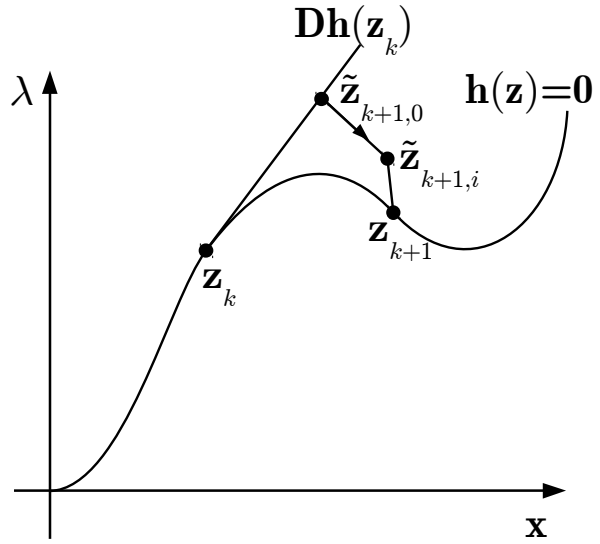


Figure 2.14: Predictor-corrector iteration with tangent prediction and correction process.

A simple approach for the prediction is to use a linear approximation of the curve $\mathbf{c}(s)$, as it is classically done in Newton-Raphson method. The predicted point is then computed by moving of a distance h along the tangent to the curve. A more sophisticated approach is proposed by Watson [120] and implemented in the *Hompack* software package. Watson uses information from previous and current iterates (position and tangent vector) to define a Hermite cubic approximation of the followed curve $\mathbf{c}(s)$. If $\mathbf{p}(s)$ is the Hermite approximation constructed on the basis of points $\mathbf{z}_k$ and $\mathbf{z}_{k-1}$ (whose curvilinear abscissa are respectively s_k and s_{k-1}), the next point on the curve $\mathbf{z}_{k+1}$ is approximated as $\tilde{\mathbf{z}}_{k+1,0} = \mathbf{p}(s_k + h)$.

The correction process of the normal flow algorithm is analogous to the procedure used in Newton-Raphson method. However, since the homotopy function $\mathbf{h}(\mathbf{z})$ is a function

from $\mathbb{R}^{n+1}$ to $\mathbb{R}^n$, its Jacobian matrix $\mathbf{Dh}$ is a rectangular $n \times (n+1)$ matrix and the Moore-Penrose pseudoinverse (see Ref. [14]) has to be used, this matrix is here denoted by $[\mathbf{Dh}(\tilde{\mathbf{z}}_i)]^\dagger$. Then the correction iteration scheme is given by the following equation:

$$\tilde{\mathbf{z}}_{i+1} = \tilde{\mathbf{z}}_i - [\mathbf{Dh}(\tilde{\mathbf{z}}_i)]^\dagger \mathbf{h}(\tilde{\mathbf{z}}_i). \tag{2.9}$$

To simplify the notations we have here omitted the curve following index k (i.e. $\tilde{\mathbf{z}}_0$ replaces $\tilde{\mathbf{z}}_{k+1,0}$). By the properties of the Moore-Penrose pseudoinverse, at each iteration, the correction $\Delta\tilde{\mathbf{z}} = \tilde{\mathbf{z}}_{i+1} - \tilde{\mathbf{z}}_i$ obtained by (2.9) is the minimum norm solution of the underdetermined problem:

$$[\mathbf{Dh}(\tilde{\mathbf{z}}_i)]\,\Delta\tilde{\mathbf{z}} = -\mathbf{h}(\tilde{\mathbf{z}}_i).$$

Equivalently, equation (2.9) involving a Moore-Penrose pseudoinverse can also be reformulated as a more usual square linear system of equations:

$$\begin{bmatrix} \mathbf{Dh}(\tilde{\mathbf{z}}_i) \\ \mathbf{v}^T \end{bmatrix} \Delta\tilde{\mathbf{z}} = \begin{bmatrix} -\mathbf{h}(\tilde{\mathbf{z}}_i) \\ \mathbf{0} \end{bmatrix}, \tag{2.10}$$

where the vector $\mathbf{v}$ corresponds to the kernel of $[\mathbf{Dh}(\tilde{\mathbf{z}}_i)]$ (assumed here to have maximum rank). As $[\mathbf{Dh}(\tilde{\mathbf{z}}_i)]$ has $n \times (n+1)$ elements, the additional line composed of $\mathbf{v}^T$ leads to a square matrix. Geometrically speaking, this line imposes the increment $\Delta\tilde{\mathbf{z}}$ to be orthogonal to the kernel of the Jacobian matrix.

The equation (2.10) explains the name *normal flow*. Indeed, for a given value of $\tilde{\mathbf{z}}_i$ the kernel vector $\mathbf{v}$ is equal to the tangent to the curve defined by:

$$\mathbf{h}(\mathbf{z}) = \mathbf{h}(\tilde{\mathbf{z}}_i).$$

If we call the *Davidenko flow* the set curves defined by the perturbed equation:

$$\mathbf{h}(\mathbf{z}) = \boldsymbol{\delta}.$$

where $\boldsymbol{\delta}$ is an arbitrary constant, one notices that consecutive correction iterates $\Delta\tilde{\mathbf{z}}$ given by (2.10) are orthogonal to the local tangent to the Davidenko flow and therefore *normal* to the so called Davidenko flow. This characteristic is represented in Figure 2.15. The solid line represent the curve $\mathbf{h}(\mathbf{z}) = \mathbf{0}$ while dashed lines are some of the curves from the Davidenko flow.

2.5.3 Implementation of the normal flow algorithm for electromechanical problems.

Aside from their original objective, algorithm developed to solve homotopy problems are also perfectly suited to solve other problems that arise in mechanics for instance in nonlinear equilibrium path tracking. Indeed, in a nonlinear buckling problem, the objective is to compute the equilibrium path into the space $(\mathbf{q}, \lambda)$. With $\mathbf{q}$ being the generalized displacements and λ the load factor. The equilibrium curve is characterized

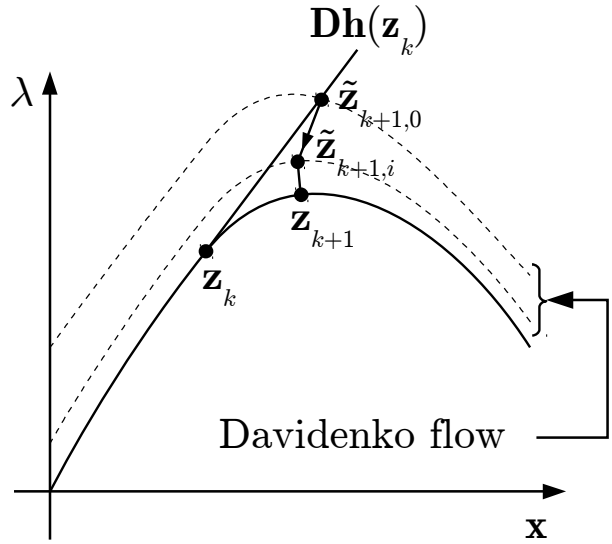


Figure 2.15: Normal flow prediction and correction process.

by the fact that external forces are balanced by internal forces in any of its points. This means that the residual forces $\mathbf{r}$ (i.e. the difference between internal and external forces) have to be equal to zero along the curve:

$$\mathbf{r}(\mathbf{q}, \lambda) = \mathbf{f}_{int} - \mathbf{f}_{ext} = \mathbf{0}. \tag{2.11}$$

In full generality, internal and external forces can depend on both generalized displacements $\mathbf{q}$ and load factor λ. For a buckling problem, the load factor only appears in the external forces which may also depend on the generalized displacements. In the scope of this thesis as explained later, no external forces are considered and the internal forces are functions of the generalized displacements and of the load factor.

From Eq. (2.11) one can easily see the analogy between on the one hand residual forces function $\mathbf{r}(\mathbf{q}, \lambda)$ and the homotopy function $\mathbf{h}(\mathbf{x}, \lambda)$ and on the other hand between the equilibrium curve and the curve $\mathbf{c}(s)$. Therefore the idea is to start from the rest position $\mathbf{q} = \mathbf{0}$ and $\lambda = 0$ (the only a priori known equilibrium point) and to follow the equilibrium curve using identical procedures as the ones that allow plotting $\mathbf{h}(\mathbf{z}) = \mathbf{0}$.

The continuation procedure has been implemented in *Oofelie* software [50, 76] in order to follow electromechanical microsystems nonlinear equilibrium path in combination with the monolithic finite element formulation that was already available in the department after the work by Rochus [91]. To get better integration with optimization and modeling tools and to improve global efficiency of the procedure, we chose to develop a specific version of the normal flow algorithm inside *Oofelie* rather than reusing existing software package like *Hompack*. The flowchart of the implemented algorithm is presented in Figure 2.16. This section is dedicated to the description of the different blocks that appear in the flowchart and details the implementation.

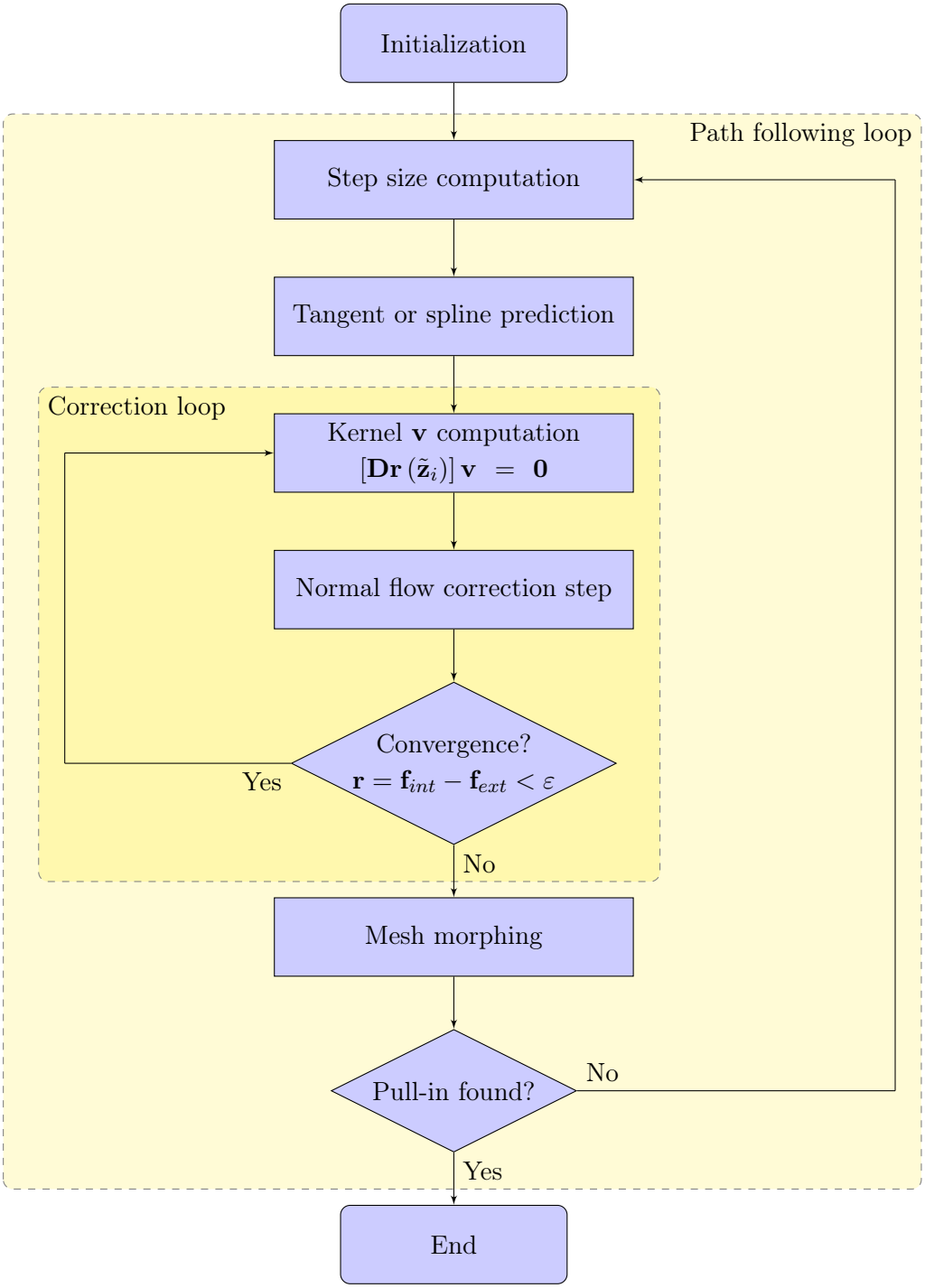


Figure 2.16: Normal flow algorithm flowchart.

Normal flow correction equation

In the case of the monolithic electromechanical formulation used in this thesis, the residual forces vector can be expressed as a function of the generalized displacements $\mathbf{q}$ and of the applied voltage V (which acts as the load variable). Moreover, in the scope of the present work the external forces are considered to be zero and the residual force vector is equal to the internal forces. Therefore, we can establish the normal flow update equation (2.10) by replacing $\mathbf{h}$ by $\mathbf{r}$:

$$\begin{bmatrix} \mathbf{Dr}\left(\tilde{\mathbf{z}}_i\right) \\ \mathbf{v}^T \end{bmatrix} \Delta\tilde{\mathbf{z}} = \begin{bmatrix} -\mathbf{r}\left(\tilde{\mathbf{z}}_i\right) \\ 0 \end{bmatrix},$$

where $\mathbf{z}$ represents now $[\mathbf{q}, V]$ and $\mathbf{v}$ is the kernel of $\mathbf{Dr}\left(\tilde{\mathbf{z}}_i\right)$. In order to implement the method in the finite element framework, the Jacobian matrix of the residual forces vector has to be split into two components:

$$\mathbf{Dr}\left(\tilde{\mathbf{z}}_i\right) = \left[\begin{array}{cc} \frac{\partial \mathbf{r}}{\partial \mathbf{q}} & \frac{\partial \mathbf{r}}{\partial V} \end{array} \right] = \left[\begin{array}{cc} \mathbf{K}_t & \frac{\partial \mathbf{r}}{\partial V} \end{array} \right]. \tag{2.12}$$

The first component corresponds to the derivative of $\mathbf{r}$ with respect to the generalized displacements vector $\mathbf{q}$. It can be replaced by the tangent stiffness matrix $\mathbf{K}_t$ as external forces are supposed to be zero. In the second component, it is worth noticing that the applied voltage corresponds to imposed degrees of freedom. To make the difference between imposed and free degrees of freedom the superscripts ^{i} and ^{f} are respectively used. For instance $\mathbf{q}^i$ denotes the generalized displacements vector containing imposed degrees of freedom while $\mathbf{q}^f$ is identical to $\mathbf{q}$ used up to now. For conciseness, ^{f} superscript is omitted whenever possible. As V only influences elements from $\mathbf{q}^i$, we can rewrite Eq. (2.12):

$$\frac{\partial \mathbf{r}}{\partial V} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}^i} \frac{\partial \mathbf{q}^i}{\partial V} = \mathbf{K}_t^{f,i} \frac{\partial \mathbf{q}^i}{\partial V}, \tag{2.13}$$

where $\mathbf{K}_t^{f,i}$ represents a submatrix of the complete tangent stiffness matrix that links free and imposed degrees of freedom. The elements of the vector $\frac{\partial \mathbf{q}^i}{\partial V}$ are constants and nonzero (generally equal to 1) if the corresponding degree of freedom is electrostatic and imposed to V, and are equal to zero otherwise.

In the end, we can write the following normal flow correction equation, which provides generalized displacements $\Delta\tilde{\mathbf{q}}$ and voltage adjustment $\Delta\tilde{V}$ to get closer from equilibrium:

$$\begin{bmatrix} \mathbf{K}_t & \mathbf{K}_t^{f,i} \frac{\partial \mathbf{q}^i}{\partial V} \\ \mathbf{v}^T & \end{bmatrix} \underbrace{\begin{bmatrix} \Delta\tilde{\mathbf{q}} \\ \Delta\tilde{V} \end{bmatrix}}_{\Delta\tilde{\mathbf{z}}} = \begin{bmatrix} -\mathbf{r}\left(\tilde{\mathbf{z}}_i\right) \\ 0 \end{bmatrix}. \tag{2.14}$$

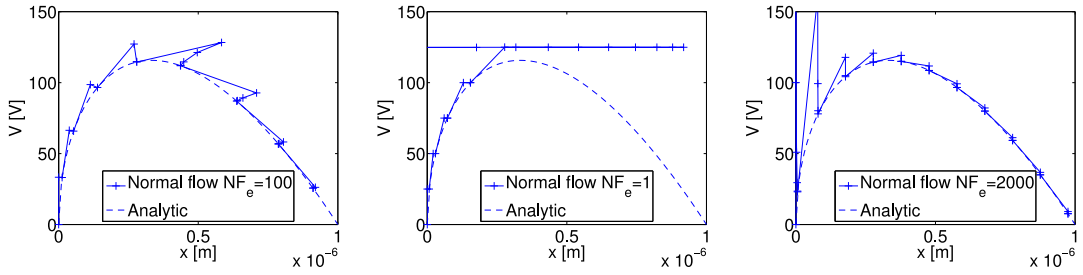
Normalization

Numerical analysis often has to resort to normalization techniques such as scaling in order to get a good conditioning of the problem and to ensure reliable and precise results.

It is also true when applying the normal flow algorithm to electromechanical microsystem models. Indeed, considering Eq. (2.14), we can see that the unknown increment vector $\Delta\tilde{\mathbf{z}}$ includes mechanical displacement degrees of freedom (of the magnitude of 10^{-6} m) and electric potential degrees of freedom (at least of the magnitude of 1 V). Because normal flow algorithm involves correction steps perpendicular to the Davidenko flow (see Figure 2.15), the relative magnitude of mechanical displacements with respect to the electric potentials has a strong influence on the correction process.

Therefore, the normal flow algorithm is applied in a normalized space where each degree of freedom is scaled according to its physical nature by using initial guesses of the pull-in point. Theoretically, only one normalization factor is needed to scale one type of degrees of freedom so that it has the same order of magnitude as the others. But for convenience, two distinct normalization factors are used, one for mechanical degrees of freedom and a second one for electrostatic degrees of freedom.

The importance of the scaling is illustrated in Figure 2.17. This Figure presents the equilibrium curve of the one degree of freedom system from Figure 2.9 solved by the normal flow algorithm for three different values of the electrostatic normalization factor (NF_e). For the three cases, the mechanical normalization factor is set to 10^{-6} m.



(a) Appropriate value. (b) Underestimated value. (c) Overestimated value.

Figure 2.17: Effect of the electrostatic degrees of freedom normalization factor.

In Figure 2.17(a), the value of the electrostatic normalization factor is close to the maximum voltage that the device can sustain and the normal flow algorithm is able to perform corrections in both directions when needed. Conversely, in Figures 2.17(b) and 2.17(c) the correction process is constrained to move along one of the axis. This result in failure to reach and pass pull-in point in Figure 2.17(b) as the algorithm is unable to reduce the applied voltage during correction process. For the present example, the choice of a too high electrostatic normalization factor is not a problem as shown by Figure 2.17(c). However, if the equilibrium curve had a local maximum in terms of displacement (snap through behavior), the situation from Figure 2.17(b) would also happen.

Step size computation

The step size determines the distance along which the equilibrium curve is approximated by a straight line or a cubic spline during prediction step. This distance is computed in the normalized space so that it remains consistent along the equilibrium curve. In the implemented algorithm, the step size is an important variable that can be tuned to improve efficiency of the continuation method or to locate accurately the pull-in point. The two tuning procedures are described below.

First, during normal continuation of the curve, step size is adapted according to the number of correction steps needed by last prediction. If too many correction iterations (more than 5) were needed to recover equilibrium the step size is reduced whereas it is increased if convergence was fast. This is a classical step size adaptation technique that helps continuation to succeed in difficult portions of the curve and to go faster in easier parts.

Secondly, step size is also adapted in order to capture accurately the pull-in point as needed by the pull-in voltage optimization procedure (see Section 4.2.2). The procedure to locate pull-in point makes use of the kernel vector $\mathbf{v}$ of $\mathbf{Dr}$ (see Eq. (2.14)) that is already computed during the correction phase. As mentioned above, when computed at an equilibrium point, this vector corresponds to the tangent to the curve. Moreover, pull-in point being a maximum of the equilibrium curve in terms of the applied voltage, the corresponding element of $\mathbf{v}$ (the last one) zeroes at pull-in point. To locate pull-in, the algorithm looks for the first zero of this last element of $\mathbf{v}$ (called $\mathbf{v}(n+1)$ here after).

The pull-in point search is disabled at the beginning of the path following and it is enabled as soon as $\mathbf{v}(n+1)$ changes sign. From that moment, we have two points of the equilibrium curve $\mathbf{z}_k$ and $\mathbf{z}_{k-1}$ for which $\mathbf{v}(n+1)$ has opposite sign. These two points surround pull-in point. Consequently, a *regula-falsi* can be applied to locate the zero point of $\mathbf{v}(n+1)$. In practice, it means that the prediction step is restarted from $\mathbf{z}_{k-1}$ (the oldest of the two points) with an updated step size h_{k-1}^{new} according to:

$$h_{k-1}^{new} = h_{k-1} \frac{\mathbf{v}(n+1)_{k-1}}{\mathbf{v}(n+1)_{k-1} - \mathbf{v}(n+1)_k}.$$

Kernel computation

In Eq. (2.14), the last element which has not been described is '$\mathbf{v}$' the kernel of $\mathbf{Dr}$. As shown by the flowchart in Figure 2.16, the kernel is computed before each correction step in order to ensure that the correction is locally perpendicular to the Davidenko flow.

In the considered problems, the matrix $\mathbf{K}_t$ has full rank over the generalized displacement space except at some specific singularity points (e.g. pull-in point or bifurcation points), that are never reached exactly. Therefore practically speaking, the matrix $\mathbf{Dr}$ has maximum rank and its kernel space has dimension 1.

Several procedures have been implemented and tested to extract $\mathbf{v}$. At first, singular value decomposition (SVD) [4] has been attempted. Since $\mathbf{Dr}$ is rectangular, it has always a singular value close to zero, the associated vector is $\mathbf{v}$. Following the same idea, one can also use the inverse power method to determine the eigenvalue closest to zero of $[\mathbf{Dr}]^T[\mathbf{Dr}]$, again the corresponding eigenvector is the searched kernel of $\mathbf{Dr}$. Alternatively, as $\mathbf{Dr}$ has maximum rank, a linear equation system solver can be used to extract its kernel as shown in the following equation:

$$\left[\begin{array}{cc} \mathbf{Dr}_{c,1} \cdots \mathbf{Dr}_{c,i-1} & \mathbf{Dr}_{c,i+1} \cdots \mathbf{Dr}_{c,n+1} \end{array} \right] \mathbf{v}_{part} = -\mathbf{Dr}_{c,i}.$$

$\mathbf{Dr}_{c,i}$ the i^{th} column of $\mathbf{Dr}$ is extracted to get a square matrix, the resulting matrix is used as the left member of an equation system while the extracted column becomes the right member. After solving this equation system the kernel of $\mathbf{Dr}$ is obtained by adding a unitary element at the i^{th} position of $\mathbf{v}_{part}$.

The three methods provide qualitatively similar results. However, the last procedure has the advantage to be faster than the two others as it involves only a linear system solution. The obvious drawback of this method is the choice of the column to extract from the matrix as the equation system becomes ill-conditioned if an inappropriate choice is done. Nevertheless, good results are obtained by choosing a column associated with a degree of freedom which moves continuously along the equilibrium curve. That's why the linear equation system method is used by default.

Mesh morphing

In certain numerical applications presented in this thesis, a portion of the domain is considered as purely electrostatic. These subdomains are void or air layers located outside of the optimization domain, they do not have any mechanical behavior. Consequently, in our framework, the most appropriate way to model such subdomains is to use electrostatic finite elements. With this approach, the nodes of electrostatic subdomains are fixed and do not possess mechanical degrees of freedom.

However, electrostatic subdomains share their boundaries with structural portions of the domain so that these boundaries are prone to move during analysis to follow deformation of the structure. This may lead to strong distortion of the electrostatic subdomains finite element mesh (as reversal of element) and to poor analysis results.

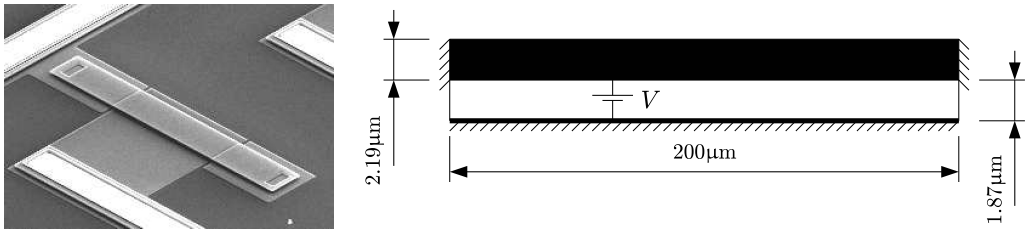
To circumvent this issue, the path following procedure includes a mesh morphing step. As proposed by Rochus *et al.* [91], the mesh morphing procedure consists in solving a fictitious mechanical problem over the electrostatic finite element mesh. The fictitious mechanical problem includes imposed displacement boundary conditions equal to the deformation of the neighbor domains. The electrostatic domain inner nodes are then repositioned according to the result of the fictitious problem. This procedure is similar to the one developed for shape optimization by Belegundu and Rajan [12].

Thanks to this procedure, the electrostatic subdomain mesh conforms to the deformation of its boundaries and the accuracy of the modeling is preserved.

2.6 Numerical application

2.6.1 Benchmark presentation

To illustrate the efficiency of the modeling procedure a numerical test case is presented in this section. The test case is inspired from an electrostatically actuated microbeam (see Figure 2.18(a)) fabricated in the framework of the ARC MEMS project[3]. The numerical test case presented in Figure 2.18(b) is a two-dimensional model of the microbeam with a simplified geometry. The ground electrode is supposed to cover the complete length of the beam and the anchors are modeled by clamping the extremities of the beam.



(a) Microbeam micrography.

(b) 2D model of the microbeam.

Figure 2.18: Microbeam test case.

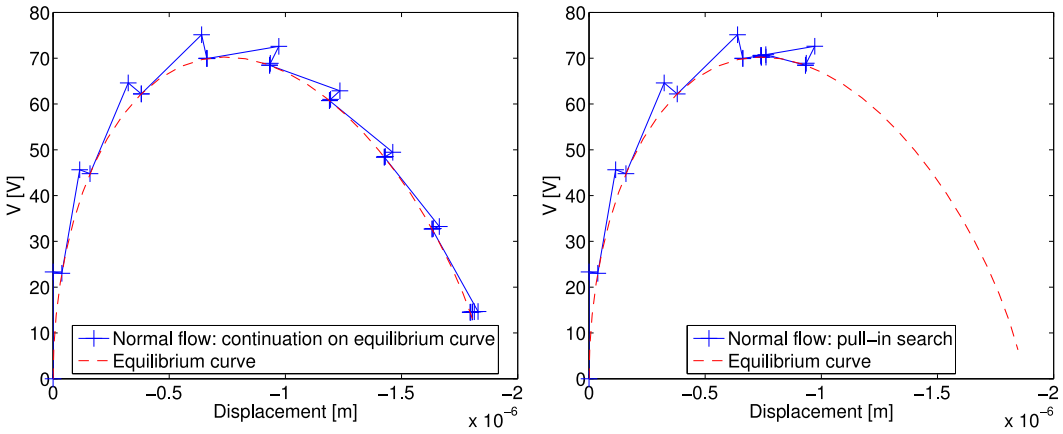
The finite element model has a total of 12000 quadrangular elements. The length of the model is divided in 800 elements, the height of the gap in 7 elements and the thickness of the beam in 8 elements. For simplicity, only electromechanical elements are used here. The beam is made of silicon with a Young Modulus of 86.79 GPa and Poisson ratio equal to 0.17. The gap is modeled with electromechanical finite elements. It is considered as void with permittivity $\varepsilon_0 = 8.85 \cdot 10^{-12}$ F/m and is assigned a very low Young Modulus (10^{-3} Pa). The mesh morphing strategy described in Section 2.5.3 is applied to adapt the mesh of the gap to the deformation of the beam.

2.6.2 Normal flow results

At first, the normal flow is executed with the objective to plot the complete equilibrium curve (i.e. without looking for pull-in point). The prediction-correction process is presented in Figure 2.19(a) together with the equilibrium curve (dashed line) obtained with a smaller step size. The abscissa corresponds to the displacement of the electrode center while applied voltage is on the vertical axis. We can identify the tangent predictions followed by correction steps to restore equilibrium.

[3]Action de Recherche Concertée 03/08-298 funded by the Communauté Française de Belgique

Secondly, the pull-in search procedure is activated. The first steps of the path following procedure are identical to the one achieved previously. However, as soon as pull-in point is passed, the procedure returns to previous equilibrium point. Then the step size is reduced using the regula-falsi procedure in order to localize pull-in point after 9 prediction-correction cycles. The pull-in voltage obtained is equal to 70.3 V and corresponds to a deflection of 0.74 µm at the center of the beam.



(a) Normal equilibrium curve continuation.

(b) Pull-in search.

Figure 2.19: Normal flow prediction and correction process for microbeam equilibrium curve and pull-in computation.

2.6.3 Comparison with approximate closed-form expression

Closed-form expressions to computing the pull-in voltage of microbeams have been derived among others by Pamidighantam *et al.* [78, 79]. In order to get an idea of the realism of last numerical results we are going to compare them with this reference. Further validation of the finite element formulation is available in Ref. [90].

The developments by Pamidighantam *et al.* are based upon pull-in voltage expression of the lumped spring-capacitor system and upon beam effective stiffness equations from Roark and Young [89]. Therefore, the pull-in voltage of a microbeam is given as a function of the effective stiffness K_{eff} and effective area A_{eff}:

$$V_{pi} = \sqrt{\frac{8K_{eff}d_0^3}{27\varepsilon_0 A_{eff}}},$$

where d_0 stands for the initial gap. In present numerical application no pre-stress is considered and boundary conditions correspond to the clamped-clamped case. Consequently, Pamidighantam *et al.* write the effective stiffness:

$$K_{eff} = \frac{32Ebh^3}{l^3\left(1-\nu^2\right)},$$

with l the length of the microbeam, h its thickness and b its width. Moreover, in the present benchmark, side effects (fringing field) are not taken into account by the 2D model (i.e. the width of the beam is supposed to be very large with respect to its thickness). Under this assumption Pamidighantam *et al.* compute the effective Area according to:

$$A_{eff} \approx \sqrt{1-\beta} bl.$$

The parameter β is equal to the normalized maximum displacement of the microbeam at pull-in. Pamidighantam *et al.* recommend to set $\beta = 0.4$ for clamped-clamped beams.

Using the geometrical dimensions of the microbeam in Figure 2.18(b), we get an estimated pull-in voltage of 72.8 V. The relative difference with respect to the value provided by the implemented normal flow procedure is of 3.6%. Therefore, we can conclude that there is a good agreement between the developed pull-in voltage evaluation procedure and Pamidighantam *et al.* closed-form expression.

2.7 Conclusion

The two modeling tools presented in this chapter are the foundations of the optimization procedures developed in this thesis. The monolithic finite element formulation is presented first. Its main assets are the inherent robustness with respect to the strong electromechanical coupling and the ability to solve both physical fields at once. This allows developing a more reliable and less complex analysis procedure.

Secondly, we choose to develop in *Oofelie* a normal flow path following algorithm to compute the equilibrium curve. According to literature, this algorithm is well suited to electromechanical coupling and to nonlinear behavior like buckling. Normal flow algorithm is able to trace the complete equilibrium curve of an electromechanical microdevice including unstable portions. This tool makes possible accurate location of characteristic points like pull-in point.

As illustrated by the numerical application, the association of the monolithic finite element formulation and the normal flow algorithm is able to provide accurate results. These results are then processed by the optimization procedure to determine design modifications that improve the response of the microdevice as explained in the next chapter.

Chapter 3

Optimization of multiphysic microsystems

3.1 Introduction

Improving performance has always been of the utmost importance even before the rise of mankind. Indeed, natural evolution leads over time to more efficient living species. Similarly, when they are facing a problem, human beings usually prefer the most efficient and most economical solution. That's why during history, increasingly powerful optimization tools have been developed. If the first optimization methods were simple and empirical, modern methods relying on mathematical foundations provide rigorous algorithms and a wide variety of application. Moreover, thanks to progress of computing power, efficient and robust optimization tools are now available. Thus, nowadays, optimization techniques are used in many fields. For instance, we can mention engineering, merchandise transportation and finance. In these domains, optimization is extensively used to guide decisions and design.

As mentioned above, engineers also resort to optimization methods. Optimization is especially used in high tech sectors such as aeronautic and space applications, where it is important to control cost, weight and now fuel efficiency. Past twenty years have seen the development of a new optimization approach more general than the previous ones i.e. *topology optimization*. Initially applied to the design of mechanical structures, the use of topology optimization is now spreading to many new areas. The application of topology optimization to electrostatic microactuators is the topic of this thesis. The detailed description of the topology optimization is the subject of the remainder of this chapter. However, as topology optimization was originally developed for mechanical applications, we will first present the classical optimization methods in mechanics in order to highlight the improvement provided by topology optimization in this area. Then in the second part of this chapter, existing applications of optimization to microsystem design will be presented.

3.2 Optimization methods for mechanical applications

The goal of optimization is to free the design process from empirical or intuitive choice of the designer. Generally speaking, an optimization method tends to be more efficient in terms of performance of the final design if it requires fewer initial inputs from the designer. Indeed, fewer inputs result in a broader design space and more freedom for the optimizer. However, increasing the generality of the optimization problem usually results in a higher computational cost. That's why, over time, optimization techniques have evolved and gained in generality concurrently with computational power.

The beginnings of structural optimization were essentially based on optimality criteria (OC). The OC method assumes prior knowledge of the conditions that design variables must satisfy to ensure optimality. An equation system whose design variables are the unknowns can then be obtained explicitly and solved. The optimality criteria have been used and developed since the sixties, the most famous one being the *Fully Stressed Design*. However, they are problem dependent and only suitable for a limited number of optimization problems such as optimization of trusses or thin structures. Also they do not always provide the exact solution of the optimization problem and lack convergence.

One difficulty of structural optimization lies in the nonlinearity and implicit character of the resulting optimization problems. Therefore, in the early sixties, Schmit [93] proposed to combine sensitivity analysis to mathematical programming algorithms in order to solve iteratively structural optimization problems. At each step of the iterative process, the derivatives of the structural responses (i.e. the sensitivities) are computed using the current analysis results. On the basis of these sensitivities, an approximate but explicit optimization problem is established and solved using mathematical programming. The new design is then analyzed once again and the iterative process is carried on until the optimum. This method has the advantage to provide faster convergence toward the optimum. Moreover, the resulting optimization algorithms are more flexible and general than those based on optimality criteria. Indeed, the only problem specific part is the sensitivity analysis. Among others the famous mathematical programming algorithms that have been applied to structural optimization are Conlin [41], MMA [105] and SQP [92].

3.2.1 Automatic sizing

Automatic sizing of structures is one of the first modern optimization technique used in mechanics. Indeed, it came right after the advent of computers and of the finite element method. Also, automatic sizing is among the first area of application of optimality criteria.

Automatic sizing can be used for optimization problems where the shape and the connectivity of the structure are known a priori. Design variables are usually chosen as the transverse dimensions of structural elements as the thickness of plates or cross sectional area of bars. If the optimization process requires several iterations before reaching the

optimum, the finite element mesh used for modeling can be kept along the optimization process as there is no geometric modification of the structure. Many studies have been dedicated to the application of optimality criteria to automatic sizing. The most popular optimality criterion is the *Fully Stressed Design* (FSD) where the optimizer tries to reach the maximum stress in each structural element. For a truss of bars of areas a_i one simple expression of this criterion is:

$$a_i^* = \frac{\sigma_i}{\overline{\sigma}},$$

which provides an updated value of the section a_i^* of bar i, with σ_i being the current axial stress within bar i and $\overline{\sigma}$ the maximum acceptable stress. This criterion allows reaching the exact solution in only one iteration for isostatic trusses. However, a hyperstatic truss requires several iterations and in this case the FSD only provides an approximation of the optimum. Other optimality criteria have been developed later to take into account several displacement constraints for instance. The interested reader may refer to the works of Berke [20], Taig and Kerr [106] and Fleury [39].

At the same time, automatic sizing has also benefited from advances in mathematical programming in the field of optimization. Indeed, as shown by Fleury [40], the combination of mathematical programming with dual methods gives rise to the generalized optimality criteria (GOC) for automatic sizing of structures.

3.2.2 Shape optimization

While automatic sizing focuses on transversal dimensions of the structural components, shape optimization has a more ambitious goal. Indeed, the objective of shape optimization is to optimize the shape of the inner or outer boundaries of a structure without changing its topology or in other words without adding or deleting holes and without changing the number of structural elements. Design variables can then be simply the geometrical dimensions of the part (length, radius ...) as presented in the example in Figure 3.1 or more generally, the control points of a NURBS curve. Shape optimization relies on mathematical programming and on sensitivity computation. Its development dates back to the seventies with the work of Zienkiewicz and Campbell [134], then it undergoes rapid evolution to finally become an industrial tool.

The main issue of shape optimization lies in the adaptation of the finite element mesh to the modifications of the boundaries of the part. Indeed, if the inner nodes remain fixed during the optimization process while the boundaries position is modified, this may result in mesh distortion and sometimes in reversed elements. To avoid this problem a first solution consists in deforming the mesh to follow the contour motion (e.g. using an auxiliary mechanical problem) with the risk after a few iterations to obtain some very ill-conditioned elements. Latter, with the emergence of automatic meshers and their increasing reliability, remeshing and error control methods have been added to the analysis process to control mesh quality and accuracy of the results [37].

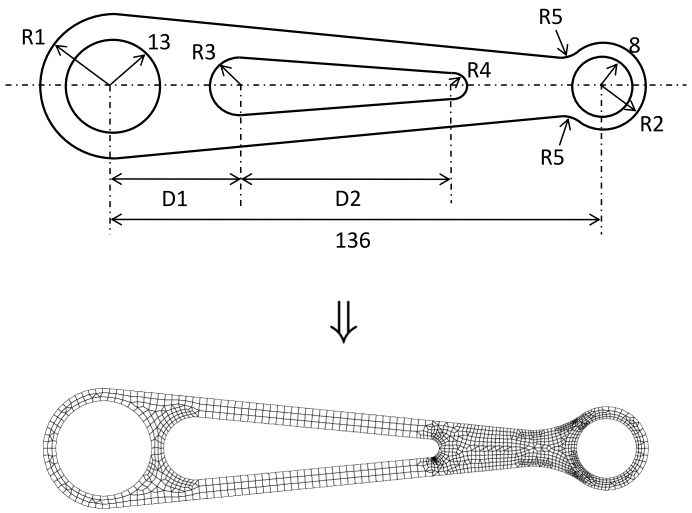


Figure 3.1: Connecting rod optimization with error control [37].

As the finite element mesh undergoes a rather complex behavior when design variables are modified, it is generally very difficult to derive analytic sensitivities for shape optimization problems. Therefore, practical applications resort (at least in part) to finite differences to compute semi-analytical sensitivities [36].

3.3 Topology optimization

Both structural optimization methods described above suffer from a common limitation which is the conservation of the topology of the optimized structure. Indeed, automatic sizing and shape optimization only apply homeomorphic transformations to the structure and are not able to modify its topology or in other words the connectivity of the domain or the neighborhood relations inside it. Because of this restriction, the optimization process result is at most an improvement of the starting structure without fundamental modification. Within this framework, the initial choices made by the designer heavily influence the outcome of the optimization process.

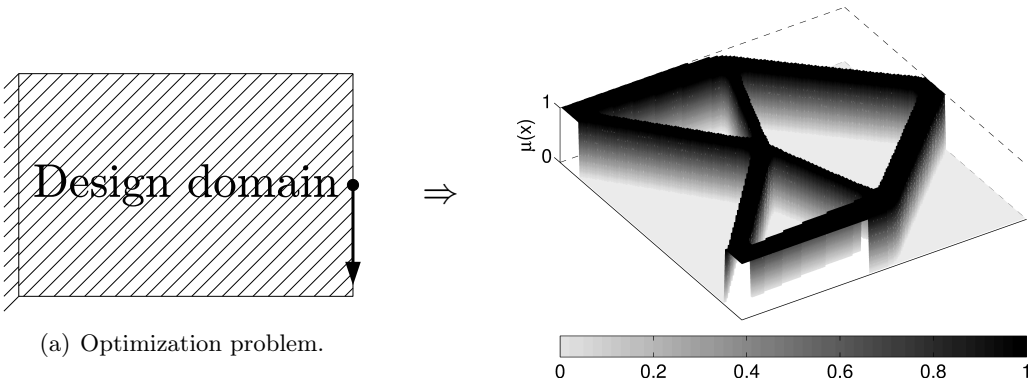
Even so, topology is one of the most important factors for the final performance. Ideally, the optimization process should be able to select by itself the optimal connectivity of the structure as well as the number of structural components composing it. To address this need, topology optimization has been developed over the last twenty years and the references [19, 38] offer a comprehensive review of many works about topology optimization.

The topology optimization problem can be formulated as the search of the optimal distribution of material in a fixed design volume. With respect to previous optimization methods, this requires a fundamental modification of the design parameterization. Explicit parameterization of the surface or contour of the structure is replaced by an

implicit representation. Two main approaches exist to implement this implicit definition of the structure. The first one considers that material distribution is described by an indicator function and gave birth to the homogenization methods. The second one came later and is based on level set functions. These two methods are described in the following sections.

3.3.1 Homogenization based methods

The most common approach consists in defining an indicator function μ over the design domain as proposed by Bendsøe and Kikuchi [17]. This function is used to indicate the presence ($\mu(\mathbf{x}) = 1$) or absence ($\mu(\mathbf{x}) = 0$) of material at some point $\mathbf{x}$ of the design domain. As illustrated in Figure 3.2, one starts by defining the design domain and boundary conditions as supports and applied forces (Figure 3.2(a)). Then the role of the optimizer is to adapt the value of the indicator function in order to distribute material over the optimization domain in such a way that the optimization problem criteria are satisfied (see Figure 3.2(b)).



(a) Optimization problem.

(b) Optimal indicator function.

Figure 3.2: Compliance minimization using topology optimization.

Considering a given material distribution, one needs to compute the response of the corresponding structure in order to evaluate optimization functions. Topology optimization classically relies on finite element method. The design domain is meshed and the indicator function μ is considered to be piecewise constant by element. Therefore, in practice, the material distribution is defined by attaching to each element a scalar value equal to the indicator function. This set of scalar values corresponds to the design variables of the optimization problem. Also, as they represent the presence or absence of material into an element they are usually called the *pseudo-densities*.

However, the purely discrete optimization problem 0-1 is a hard problem and even if recent advances show promising results (see [72]), the most versatile approach consists in relaxing the optimization problem by switching to continuous variables as proposed by Bendsøe and Kikuchi [17]. That's where the term *pseudo-density* makes complete

sense since the design variables are allowed to vary continuously from 0 to 1, representing porous material or simply being an artificial interpolation to avoid the discrete problem. This conversion allows using sensitivity analysis and resorting to efficient continuous optimization algorithms for optimization problem solution.

The pseudo-densities are used to define the properties of each finite element. In the discrete case, the relation between pseudo-densities and the element properties is straightforward. For instance, the elastic modulus is set to the design material one if pseudo-density is equal to 1 and set to a very low value (very soft material) if the pseudo-density is equal to 0. In the continuous case, the situation is more complicated since one needs to compute element properties for intermediate pseudo-densities.

Intermediate pseudo-density representation

Microstructural approach

Seminal work by Bendsøe and Kikuchi [17] proposes to use a periodic porous material to represent intermediate pseudo-densities. The microstructure they have defined is represented in Figure 3.3. The microstructure is obtained by creating rectangular microperforations into an isotropic material. Instead of one single pseudo-density variable, the microstructure is defined by three parameters. The relative size of the microperforations is determined by two parameters α_1 and α_2 while θ defines the orientation of the microstructure. Consequently, this approach results in 3 design variables per element. Eventually, the indicator function can be easily computed using α_1 and α_2.

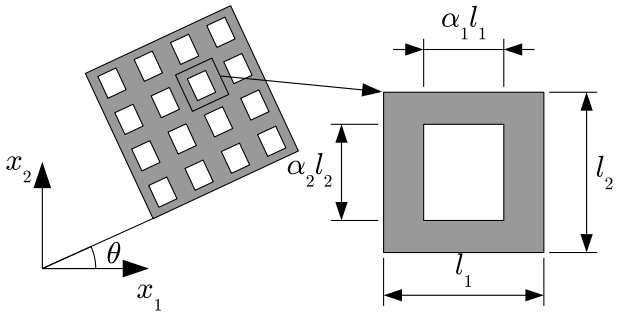


Figure 3.3: Sketch of the microstructure proposed by Bendsøe and Kikuchi [17].

For a given set of parameters α_1, α_2 and θ, it is then possible using homogenization theory to compute the equivalent Hooke tensor of a macroscopic sample of the material. These computations rely on a finite element model of the unit cell.

The advantage of the microstructural approach lies in its physical relevance. The optimal structure behaves in a physically correct way even if the design domain is not free from intermediate pseudo-density elements. That's why several microstructures have been studied along the development of topology optimization. As another example,

we can mention the layered materials (see [15]) for which analytical expressions of the equivalent properties are available.

Solid Isotropic Material with Penalization (SIMP)

The use of homogenized microstructures to represent intermediate pseudo-densities of the relaxed topology optimization problem has two drawbacks from a practical point of view. Firstly, the microstructural approach has the disadvantage of providing optimal topologies with a lot of intermediate pseudo-density elements. If microstructures are theoretically convenient, they are difficult to manufacture, which makes optimization results with intermediate pseudo-densities difficult to produce in practice. Secondly, the computational cost required by the homogenization procedure leads to the search for analytical and explicit laws linking pseudo-density to material properties. The most famous material model is the SIMP model (Solid Isotropic Microstructure with Penalty) proposed by Bendsøe [15]. This law scales the actual Young modulus of the design material E_s by a power function of the pseudo-density to compute the pseudo-material Young modulus E, while the density is interpolated linearly:

$$\begin{gathered} E(\mu) = \mu^p E_s, \\ \rho(\mu) = \mu \rho_s, \\ 0 \leqslant \mu \leqslant 1 \quad \text{and} \quad p > 1. \end{gathered}$$

Because parameter p is chosen greater than 1, the stiffness E provided by non-integer pseudo-densities is weak with respect to the resulting material resource consumption ρ which is interpolated linearly as illustrated in Figure 3.4. Consequently, when the optimization problem includes a limit on the available amount of material, the SIMP law penalizes the use of intermediate densities and the optimizer tends to gather material in order to end up with a 0-1 distribution. The elimination of intermediate densities is extremely interesting in practice since it gives rise to a structure, which is much

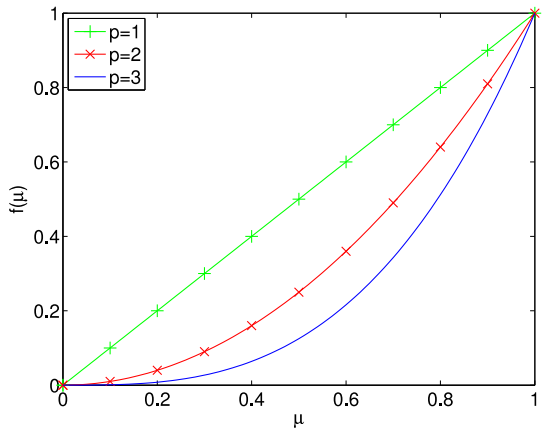


Figure 3.4: SIMP function plot for three different values of parameter p.

more realistic and easier to interpret in order to manufacture it. This trend combined with the simplicity of the model is the origin of the success of the SIMP interpolation. Usually, an exponent value equal to 3 or 4 is enough to reach a 0-1 distribution.

Despite the artificial basis of the SIMP law, Bendsøe and Sigmund [16] have shown that it is possible to design a microstructure which provides the same macroscopic behavior as the one predicted by the SIMP law. However, Bendsøe and Sigmund demonstrate that there is a minimum value of p below which it is not possible to find a corresponding microstructure. This minimum threshold depends on the Poisson ratio ν of the design material, for $\nu = 1/3$ the minimum value of p is 3.

Halpin Tsai - RAMP law

From its original introduction, SIMP method has been used successfully in many topology optimization applications. However, the need for alternative material model arises as soon as mass inertia has been involved in the physics underlying the optimization problem. Indeed, when attempting eigenfrequency optimization [83] or problems involving self-weight [27] troubles happen for low density regions in which residual stiffness becomes nearly zero while the mass remains non-zero. This results in meaningless local eigenmodes.

As shown by Bruyneel and Duysinx [27] the major property to circumvent the difficulty is to keep a non-zero ratio between the stiffness and the mass when the density tends to zero. In other words, that means that the slope of the stiffness interpolation function must be different from 0 for low pseudo-density which is not the case for the SIMP law (for $p > 1$).

The work by Halpin and Tsai [44] allows obtaining an interpolation law that fulfills this requirement. Halpin and Tsai propose semi-empirical homogenization laws to predict effective stiffness properties of fiber composites. Considering a material of Young modulus E_s perforated by parallel fibers made of a very weak material, the equivalent Young modulus E in the orthogonal plane to the fiber is given by:

$$E(\mu) = \frac{\mu\xi}{1 + \xi - \mu} E_s. \tag{3.1}$$

The variable $\xi > 0$ reflects the reinforcement (or weakening) of the matrix by the fibers. In topology optimization context, ξ can be used as a penalization parameter, $\xi = \infty$ corresponds to a linear interpolation and penalty increases when ξ is decreased as illustrated in Figure 3.5. Also, Figure 3.5 allows comparing Halpin Tsai law to SIMP material model ($p = 3$). One can see that for small pseudo-densities, Halpin Tsai law provides more stiffness than SIMP even for high value of penalty ($\xi = 0.1$). Thanks to this characteristic Halpin Tsai law provides better behavior in optimization problems like eigenfrequencies optimization. However, this comes to the cost of a less efficient penalization of low pseudo-densities and leads to slightly less contrasted topologies.

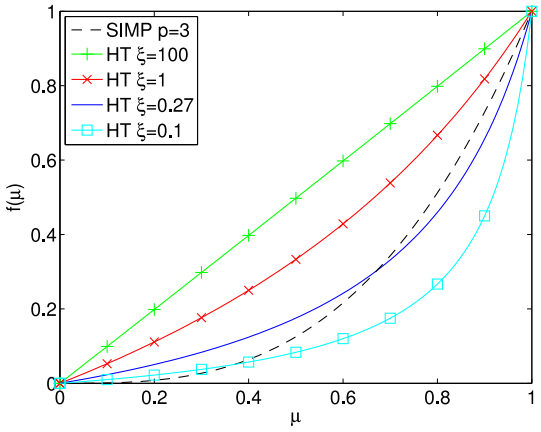


Figure 3.5: Halpin Tsai function plots for different values of parameter ξ and compared to SIMP law.

Halpin Tsai law is actually very similar to RAMP (Rational Approximation of Material Properties) material model developed by Stolpe and Svanberg [104] in order to obtain particular convexity/concavity properties for the compliance function. RAMP model covers a more general case in which it is possible to define a minimum Young modulus E_{min} that is obtained for $\mu = 0$. The original expression of RAMP law is:

$$E(\mu) = E_{min} + \frac{\mu}{1 + q(1-\mu)}(E_s - E_{min}),$$

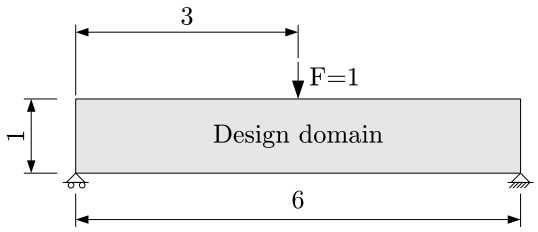
with q being the penalty parameter. With very simple algebraic manipulations this expression can be transformed as follows:

$$E(\mu) = E_{min} + \frac{\frac{\mu}{q}}{1 + \frac{1}{q} - \mu}(E_s - E_{min}).$$

Which is equivalent to Eq. (3.1) if E_{min} is set to zero and $\xi = 1/q$.

Problem regularization

Previous sections describe how it is possible to transform the discrete 0-1 topology optimization problem into a continuous variable problem which is easier to solve. However, the enlargement of the design space to non-entire densities is not sufficient in general to obtain a well-posed optimization problem. Indeed, the solutions of the relaxed optimization problem usually suffer from mesh dependency and from the apparition of checkerboard pattern. Mesh dependency of the solution is illustrated in Figures 3.6 for the compliance optimization of the so called classical MBB beam [75] presented in Figure 3.6(a). Figure 3.6(b) and Figure 3.6(c) shows clearly that reducing elements size by a factor two results in a very different topology involving thinner substructures. This tendency to create thinner structure with mesh refinement is a consequence of the



(a) MBB beam topology optimization problem.

(b) 120 × 20 finite element mesh.

(c) 240 × 40 finite element mesh.

Figure 3.6: Influence of the finite element mesh on the final design.

topology optimization problem ill-conditioning as the solution will never converge to a stable design if the finite element mesh is refined.

The second difficulty arising in topology optimization is the apparition in the final design of zones covered by an alternation of void and solid elements similar to a checkerboard as it is illustrated in Figure 3.7. Diaz and Sigmund [33] and Jog and Haber [47] have shown that the stiffness of the checkerboard pattern is overestimated by the finite element method especially while using first order elements. Its stiffness is numerically higher than the one obtained with uniform distribution of the same amount of material. As a result, the checkerboard pattern is artificially preferred by the optimizer. This numerical artifact makes difficult the interpretation of the final design and therefore has to be prevented.

Figure 3.7: Example of checkerboard pattern.

Different solutions have been developed in order to regularize the optimization problem and by the way to avoid mesh dependency and checkerboard. Sigmund and Peterson [102] have proposed a review of most of the existing regularization methods as perimeter constraint, density gradient constraints and sensitivity filtering. We can add to this list density filtering that has been introduced later by Bruns and Tortorelli [25]. Out of these methods, we focus in the following of this section on the description of filtering techniques because these are the ones who had the most success.

Sensitivity filter

Sensitivity filtering has been proposed by Sigmund [98] and consists in using a filter technique inspired from image processing to smooth out the sensitivity field. In spite

of the heuristic nature of sensitivity filter, it has shown to provide good results and is widely used in currently available commercial topology optimization software. The modified sensitivities $\partial\hat{f}/\partial\mu_i$ are then computed as follow,

$$\frac{\partial \hat{f}}{\partial \mu_i} = \frac{\sum_{j=1}^{N} H_{ij}\mu_j \frac{\partial f}{\partial \mu_j}}{\mu_i \sum_{k=1}^{N} H_{ik}} \quad \text{with} \quad H_{ij} = \max\left(0, R - \text{dist}\left(i,j\right)\right). \tag{3.2}$$

The function $\text{dist}\left(i,j\right)$ provides the distance between finite elements i and j centroids. Consequently, the sensitivity of each element is replaced by a weighted average of the sensitivities of elements included in a neighborhood of radius R. This way, sensitivity filter prevents the appearance of checkerboard patterns. Moreover, the filter introduces a minimum size constraint on the structural members which is very interesting from a manufacturing point of view. This minimum size can be adjusted by modifying the radius R. Therefore, when refining the finite element mesh, keeping the filtering distance constant prevents the creation of thinner members and avoids the modification of the optimal layout. The effect of sensitivity filter method is illustrated in Figures 3.8 where the filters radii are both equal to $R = 0.06$ but the mesh is twice thinner in Figure 3.8(b) than in Figure 3.8(a).

(a) 120 × 20 finite element mesh. (b) 240 × 40 finite element mesh.

Figure 3.8: Regularization by sensitivity filtering with $R = 0.06$.

Later, the ability of the filter to introduce a restriction on the smallest size of structural members has been improved by Zhou *et al.* [132] by modifying the SIMP penalty parameter during the optimization process.

Density filter

More recently, another filter based regularization method has been proposed by Bruns and Tortorelli [25]. Bruns and Tortorelli propose to make a distinction between the design variables and the elementary pseudo-densities. They introduce a new set of design variables x_i distributed over the optimization domain (for instance one per element) and compute the pseudo-densities μ_i as a weighted average of the neighbor design variables. The expression proposed by Bruns and Tortorelli to calculate the density is in fact similar to the sensitivity filter and is given by,

$$\mu_i = \frac{\sum_{j=1}^{N} H_{ij} x_j}{\sum_{k=1}^{N} H_{ik}} \quad \text{with} \quad H_{ij} = \max\left(0, R - \text{dist}\left(i,j\right)\right), \tag{3.3}$$

where the operator $\text{dist}\left(i,j\right)$ represents the distance between the element i barycenter and the location of the variable x_j (generally the element j barycenter). The sensitivities

evaluation is then a little more sophisticated since a perturbation of the design variable x_i influences several pseudo-densities. Therefore, for each variable x_i, we have to apply the following chain rule:

$$\frac{df}{dx_i} = \sum_j \frac{\partial f}{\partial \mu_j} \frac{d\mu_j}{dx_i}.$$

However, Bourdin [22] proved that density filtering gives rise to a well-posed optimization problem. Moreover, the non-heuristic character of this method is its principal advantage on sensitivity filtering while the parameter R also allows introducing a minimum size constraint. The effect of the filter is presented in Figure 3.9 which again shows that a refinement of the mesh does not influence the final topology if the radius of the filter remains the same ($R = 0.085$).

(a) 120 × 20 finite element mesh. (b) 240 × 40 finite element mesh.

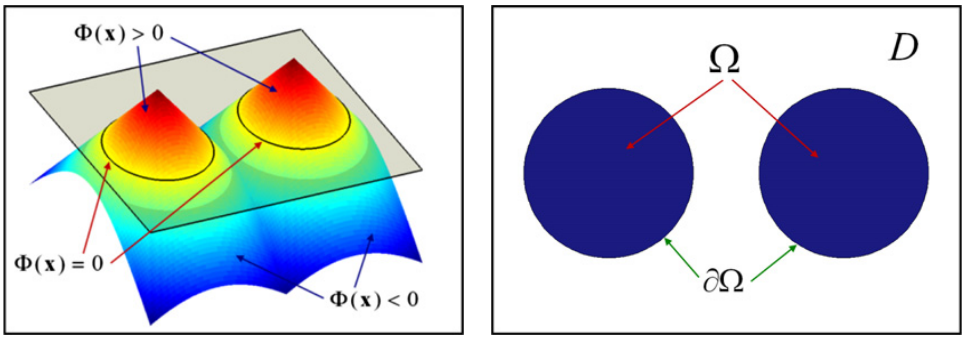
Figure 3.9: Regularization by density filtering with $R = 0.085$.

One drawback of this filter is that the resulting topologies possess very smooth boundaries between void and solid resulting in grey regions. A few possibilities have been studied to remove grey material from the final design as bilateral filtering by M.Y. Wang and S. Wang [116] and image morphology operators by Sigmund [100].

3.3.2 Level set method

Aside from homogenization related methods, another approach based on a level set representation of the design is gaining interest. The level set method introduced by Osher and Sethian [77] consists in representing the boundary of the structure through a higher dimensional function $\phi(\mathbf{x})$ defined over the whole optimization domain. In some sense, the function $\phi(\mathbf{x})$ is similar to the indicator function described in the homogenization approach description. However, in level set method the boundaries are implicitly defined by the equality $\phi(\mathbf{x}) = 0$. The curve or the surface defined by this equation separates the structural part of the optimization domain from the non-structural part. This is illustrated in Figure 3.10 for a 2D design domain D. As shown in Figure 3.10(a), the corresponding level set function is a 3D surface and the contour defined by its intersections with the zero level plane defines the boundaries of the structure presented in Figure 3.10(b). Luo *et al.* [66] consider that points for which $\phi(\mathbf{x}) > 0$ belong to the structure while the places for which the level set surface is negative belong to void area of the optimization domain. However, the inverse convention is also used by other researchers (see Allaire *et al.* [3] for instance).

In the optimization context the most important advantage of the level set representation is its transparency to topological changes (i.e. holes can be added, merged or removed



(a) 3D level set surface. (b) 2D boundary at zero level set.

Figure 3.10: Implicit boundary representation using level set function [66].

without requiring a new parameterization of the optimization problem). As suggested by Osher and Sethian [77], the evolution of the level set function across optimization is generally governed by the Hamilton-Jacobi equation:

$$\frac{\partial \phi}{\partial t} + v\nabla\phi = \mathbf{0}, \tag{3.4}$$

where v can be interpreted as a speed function. It represents the requested displacement of the surface computed by the optimization procedure in order to improve the design.

The first application of level set method into an optimization procedure is due to Sethian and Wiegmann [95]. Sethian and Wiegmann developed a procedure that allows improving the rigidity of a structure using an adequate criterion based on the Von Mises stresses. They used an immersed interface method to compute the stresses and take into account the structural boundaries defined by the level set. The modification of the level set function is ensured by solving the Hamilton-Jacobi equation (3.4) using a finite difference procedure (upwind scheme).

Later, to avoid complexity and restrictions brought by the immersed interface procedure, a so called *ersatz material* approach has been proposed by Allaire *et al.* [3] and M.Y. Wang *et al.* [117]. The method relies on finite element method to solve the mechanical problem. The finite element mesh covers the complete optimization domain but a very weak material (i.e. the ersatz material) is affected to non-structural elements. Moreover, as shown by Allaire *et al.* [3], finite element method allows computing a classic shape derivative of the design. This shape derivative is then used as the speed function in the Hamilton-Jacobi equation.

However, solving the Hamilton-Jacobi equation may leads to numerical difficulties. Therefore, alternative level set methods that avoid solving Hamilton-Jacobi equation have been developed. One idea is to substitute the implicit definition of the level set function by an explicit definition. For instance, Van Miegroet and Duysinx [112] define the level set function using geometric primitives whose parameters are the design variables of the optimization problem. Moreover, modeling is carried out using the

extended finite element method which avoids using ersatz material. Another approach consists in resorting to a nodal definition of the level set function. The nodal values of the level set are then directly used as design variables of the optimization problem (see for instance Van Dijk *et al.* [109]). Similarly, Luo *et al.* [66] have proposed to use a sum of radial basis functions distributed over the optimization domain to define the level set function.

The application of the level set method to structural optimization is very promising and is gaining popularity. However, it has not yet reached the generality of the homogenization approach of topology optimization. Indeed, even if the level set method allows topological modifications as merging of entities, insertion of new entities (like holes) appears to be more difficult. The development of methods that provide topological derivative (i.e. sensitivity to the introduction of a small hole at one point) is still in progress (see for instance Amstutz *et al.* [7] and Novotny *et al.* [8]).

3.4 Shape optimization of electrostatic microsystem

3.4.1 Pull-in voltage optimization

Microbeam pull-in voltage shape optimization has been proposed by Abdalla *et al.* [1]. Their objective is to maximize pull-in voltage of a microbeam by modifying its thickness or width profile. Several support conditions are studied. The sketch of the microsystem studied by Abdalla *et al.* is presented in Figure 3.11.

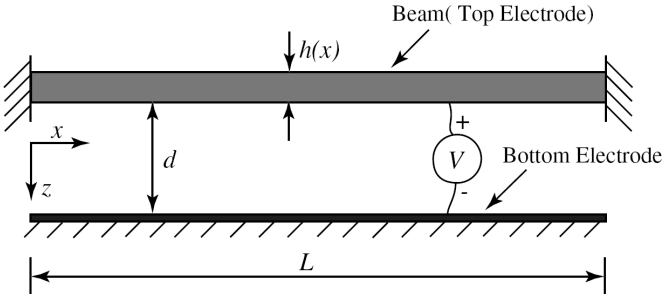


Figure 3.11: Sketch of the optimized microsystem [1].

Electromechanical modeling

Abdalla *et al.* use beam finite elements to model the microsystem. Each element owns width and thickness which are supposed to be constant over the element. The electrostatic force is computed as a function of the applied voltage V and of the distance between the two electrodes. By neglecting side effects, we get the following expression of the electrostatic force f_{es} applied at abscissa x:

$$f_{es}(x) = \frac{\varepsilon_0}{2} \frac{b(x) V^2}{(d - w(x))^2}. \tag{3.5}$$

The width of the element $b(x)$ is of course included in this expression since it influences the surface thereof. Conversely, the thickness of the beam $h(x)$ is not taken into account in this equation. Even if h can also vary along the beam, the gap is assumed to be constant regardless of the thickness distribution. Therefore, the distance that separates the two electrodes is computed as the difference between the initial gap d and the displacement of the beam $w(x)$. Pull-in point is computed using the finite element model and the *normal flow* algorithm [85] whose implementation is similar to the one presented in Section 2.5.1.

Optimization problem

The goal of the optimization process is to determine the optimal thickness and width distribution in order to maximize pull-in voltage. Therefore, the optimization problem can be formulated as follow:

$$\max_{y(x)} \lambda_{cr},$$
$$\text{s.t.} \begin{cases} \int_0^1 y\, dx = 1, \\ \underline{y} - y(x) \leqslant 0. \end{cases}$$

Objective function λ_{cr} corresponds to the dimensionless critical pull-in load. It is proportional to the square of pull-in voltage. Design variables $y(x)$ correspond either to the dimensionless thicknesses $\frac{h(x)}{h_0}$, or to dimensionless widths $\frac{b(x)}{b_0}$. Two constraints are included in the optimization problem. First of all, a constraint imposes the volume of the beam to be equal to the initial volume. Secondly, a lower bound over design variables allows taking into account manufacturing requirements on the smallest manufacturable width or thickness. Optimization is performed using an optimality criterion derived from the expression of the Lagrangian of the optimization problem.

Numerical applications

Starting from a uniform initial distribution $y(x) = 1$, Abdalla *et al.* have tested their optimization procedure for thickness and width optimization separately. Figure 3.12 presents the results of thickness optimization while Figure 3.13 those obtained with

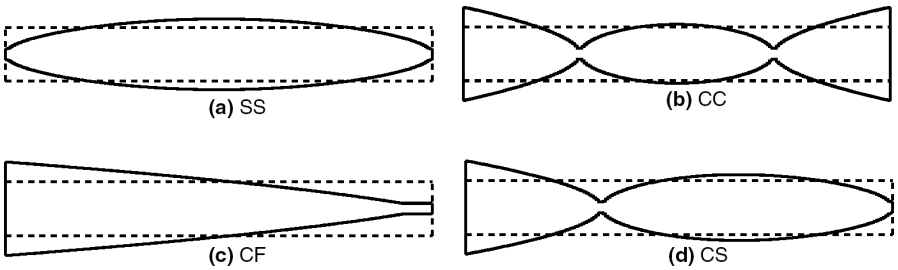


Figure 3.12: Thickness distributions for different boundary conditions [1].

width optimization. In these figures, the initial configuration, which corresponds also to the fixed material volume constraint, is sketched as a dashed line.

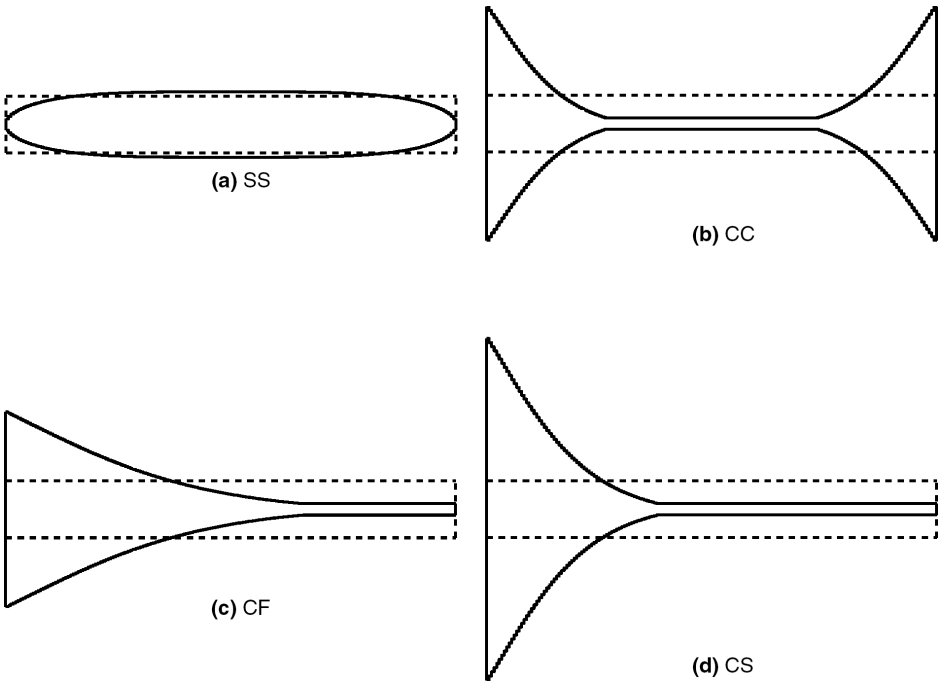


Figure 3.13: Width distributions for different boundary conditions [1].

The thickness optimization examples proposed in Figure 3.12 are computed with a lower bound on the thickness $\underline{h} = 0.2$. Table 3.1 compares the value of λ_{cr} before and after optimization. Notice that for almost all configurations the improvement is greater than 50%. The analysis of the optimal shapes in Figure 3.12 shows that the optimizer tends to adapt material distribution in order to maximize the stiffness of the microbeam.

	SS	CC	CF	CS
Initial	1,1492	5,8413	0,1401	2,7880
Final	1,700	10,0136	0,3391	4,4405
Improvement (%)	47,93	71,32	142,06	59,27

Table 3.1: Improvement of λ_{cr} with thickness optimization [1].

Width optimization gives the optimizer an additional degree of freedom since it can also change the distribution of electrostatic forces on the structure. Indeed, modification of the width of an element implies a variation of the electrostatic force applied on this element (see Eq. (3.5)). Consequently, as shown by Figure 3.13, the optimizer gathers material close to the clamped extremities in order to minimize lever arm of electrostatic force while continuing to seek maximum stiffness. Thanks to the possibility to move electrostatic forces, the improvement of λ_{cr} is larger than previously and can reach over 400% for the clamped-free case.

3.5 Topology optimization of microsystems

3.5.1 Electrothermal actuator optimization

Like electrostatic forces, electrothermal forces are frequently used in microsystems. Electrothermal actuators operate using heat generated by an electric current passing through a portion of the device. As a result, the temperature of the portion submitted to the electric current heats up and expands. Topology optimization of such microactuators has been investigated by Sigmund [99], Yin and Ananthasuresh [128], Mankame and Ananthasuresh [69] and Luo *et al.* [66].

Of course, the optimization problem is multiphysic since three physical phenomena occur simultaneously i.e. electric conduction, heat transfer and mechanical deformation. Nevertheless, the coupling between these phenomena is purely sequential and one-way as illustrated in Figure 3.14. Therefore, numerical modeling of the device can be performed

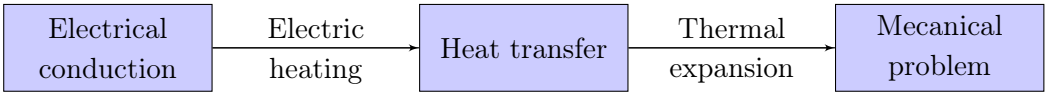


Figure 3.14: Electrothermal actuator coupling sequence.

using a staggered procedure without requiring iteration. Each physical problem is solved in logical order.

Sigmund [99] and Yin and Ananthasuresh [128] consider the problem of maximizing the displacement u_{out} of one point of the optimization domain that we call the output port. The typical optimization problem includes a constraint on material resources as well as lower and upper bound on design variables:

$$\max_{\boldsymbol{\mu}} u_{out}(\boldsymbol{\mu}),$$
$$s.t. \begin{cases} v(\boldsymbol{\mu}) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i. \end{cases}$$

The interpolation of material properties between void and solid is performed for each physical field using a power law. Both papers also consider the design of bi-material microsystems. In bi-material actuator, the difference between expansions coefficients of the two design materials enables reaching better performance.

Moreover, Sigmund shows that the optimization procedure can be used with several 'load cases'. These load cases differ by the position of the electrodes and by the requested direction of the output displacement. An example is presented in Figure 3.15. For the first load case, it was requested that an input voltage at electrode V_1 results in a horizontal displacement. Conversely for the second, an input voltage at electrode V_2 must give rise to a vertical displacement of the output port.

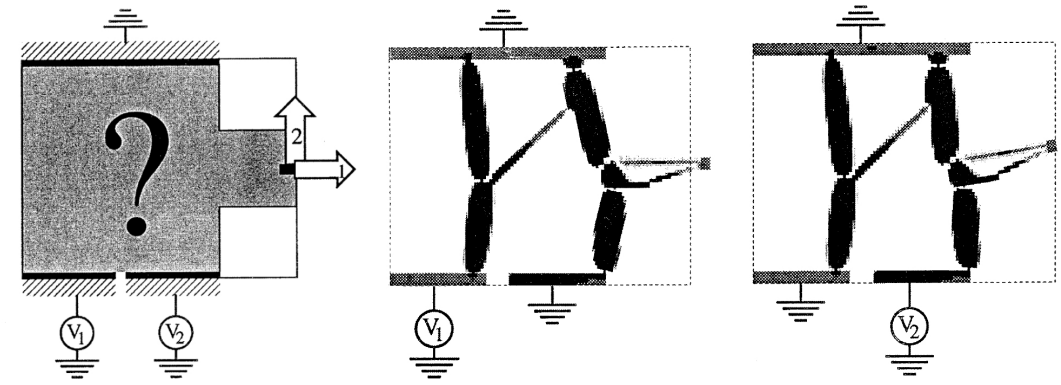


Figure 3.15: Optimization problem and results by Sigmund [99] with several load cases.

Unlike the *continuum media* topology optimization formulation used by Sigmund and Yin, Mankame and Ananthasuresh [69] choose to study optimization of an electrothermal actuator based on a bar truss structural universe. Except the advantage of an optimal structure usually clearer than with continuum media topology optimization, the interest of bar elements is that they allow simpler modeling of electric phenomena and of heat transfer. However, the resulting optimization problem is more constrained on a geometrical point of view since the locations of the structural members are fixed a priori. Figure 3.16 shows one of the applications treated by Mankame and Ananthasuresh. The sketch on the left presents the fixation points, the input point of the two electrodes and the output port. The deformed optimal layout is presented in the right figure. In the right figure, the dashed lines correspond to bar elements that reached their lower bound so that they do not participate to the structure.

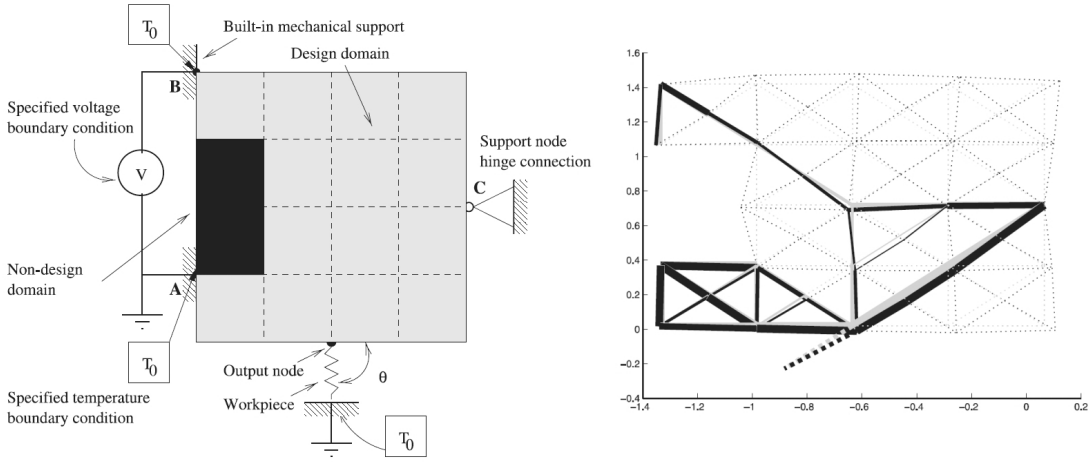
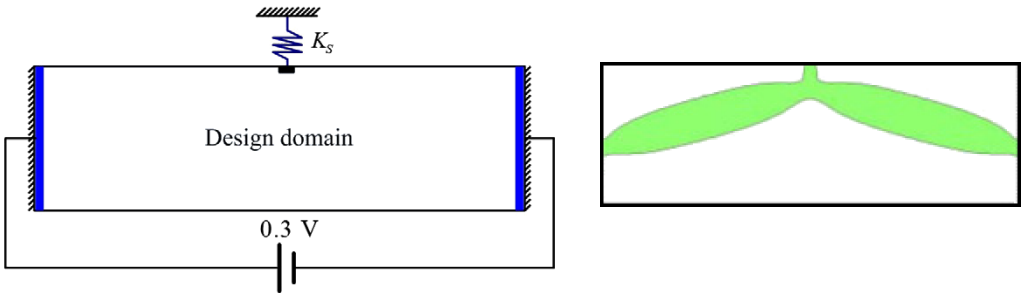


Figure 3.16: Truss design problem and result by Mankame and Ananthasuresh [69].

Recent work by Luo *et al.* [66] investigates electrothermal actuator optimization using level set representation. The level set function is defined explicitly as a sum of radial

basis function distributed over the optimization domain. The design parameters are the amplitudes of the radial basis functions. Moreover, the modeling procedure used by Luo *et al.* relies on a meshless Galerkin method and is able to handle geometric nonlinearity. Conversely to the mechanical field, thermal and electrical physical field are considered as linear. Using the developed method, Luo *et al.* are able to design simple electrothermal actuators as presented in Figure 3.17.



(a) Optimization problem. (b) Optimal actuator.

Figure 3.17: Electrothermal actuator design by Luo *et al.* [66].

Alternatively, Langelaar *et al.* [56] propose to replace linear materials with shape memory alloys (SMAs). Compared to classical materials, SMAs can provide larger strains for a given temperature variation. However, their behavior is highly nonlinear such that complex actuators cannot be designed without the help of appropriate modeling tools. Therefore, Langelaar *et al.* develop a topology optimization procedure able to design thermal actuators. The objective is to maximize the output displacement difference between two given temperatures. Even though the electrical heating and thermal conductivity are not yet modelled, the proposed model can be enriched with other physical fields.

3.5.2 Electrostatic actuator topology optimization

Staggered approach

The work of Abdalla *et al.* [1] presented in Section 3.4 consists in applying shape optimization to microdevices that exhibit electromechanical coupling. However, Raulli and Maute [86] have shown that it was also possible to adapt topology optimization to tailor these multiphysic devices.

Electromechanical modeling used by Raulli and Maute is based on a staggered method. This means that the two physical fields (electrostatic and mechanical) are solved separately. Because of the strong coupling between electrostatic and mechanical fields, they have to resort to iteration to restore equilibrium.

Modeling of variable material distribution

Generally speaking, the modeling of an electrostatically actuated microsystem can be performed in a staggered way by meshing separately the mechanical structure and the surrounding void space (i.e. the electrostatic domain). Both physical fields are then solved on their respective mesh.

However, when considering a topology optimization problem, it is not possible to mesh individually the two domains. Indeed, at each iteration, the topology of the physical domains can be modified. Furthermore, when topology optimization problem is relaxed to the non-integer pseudo-densities, it is difficult to clearly separate the two physical domains because of the presence of intermediate density regions.

In order to circumvent these two issues, Raulli and Maute propose to extend the electrostatic mesh to the entire design domain. Figure 3.18 shows the superposition of the meshes, with Ω_0 representing the purely electrostatic domain in which the optimization process is not allowed to place material and Ω_δ being the part of the electrostatic mesh that covers the optimization domain.

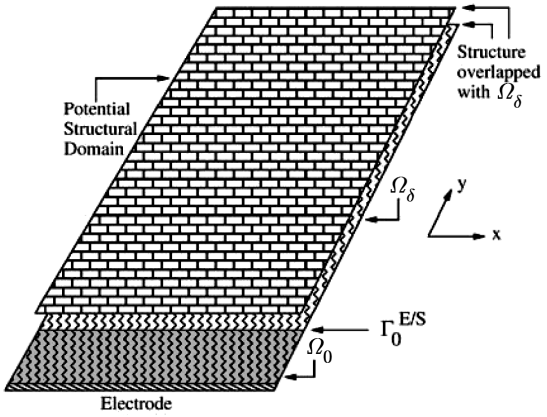


Figure 3.18: Sketch of the computational domains superposition [86].

Of course, the results of the electrostatic computation depend on the material distribution defined by the optimizer. Similarly to mechanical topology optimization, the effect of the material distribution on the electrostatic domain is represented by modifying the dielectric permittivity of each element using its pseudo-density. The design material is assumed to be a perfect conductor, therefore a very high permittivity ε_{max} (infinite in theory) is associated with $\mu = 1$ while void permittivity $\varepsilon_0 = 8.85\,10^{-12}$ is affected to element for which $\mu = \mu_{min}$. Therefore, the material model is the following:

$$\begin{cases} E(\mu) = \mu^{p_E} E_s, \\ \varepsilon(\mu) = \varepsilon_{max}(\mu - \mu_{min})^{p_\varepsilon} + \varepsilon_0 \quad \text{with} \quad \varepsilon_{max} = \frac{\varepsilon_0}{\mu_{min}^{p_E}}, \end{cases} \tag{3.6}$$

with p_E and p_ε being the penalty parameters respectively on Young Modulus and permittivity. The permittivity ε_{max} is computed such that its ratio with respect to ε_0 is equal to the ratio between E_s and $E(\mu_{min})$ which means equal to $\mu_{min}^{-p_E}$.

Imposed voltage and electrostatic forces

Two difficulties remain in the present modeling procedure, first the definition of the input voltage and second the computation and application to the structural domain of the electrostatic force. These two boundary conditions should normally be applied on the movable boundary between structural and electrostatic domains. However because of intermediate pseudo-densities, this boundary is blurred and not accurate especially at the beginning of the optimization process. The solutions proposed by Raulli and Maute enable imposing those boundary conditions even in presence of intermediate pseudo-densities. These solutions are, under some aspects, artificial and they complicate the physical understanding of the modeling. However, the application presented at the end of this section shows the efficiency and the interest of their method.

To impose the input electric potential over the structural part of the domain, a modification is introduced into the electrostatic problem equation:

$$\mathbf{K}_{\phi\phi}\boldsymbol{\phi} = \mathbf{Q}_{es}.$$

This equation allows computing the electric potential vector $\boldsymbol{\phi}$ knowing the electrostatic charges vector and the permittivity matrix. To prescribe the input voltage at one node, a perturbation is applied to its corresponding element in the diagonal of matrix $\mathbf{K}_{\phi\phi}$ as well as the associated electric charges in vector $\mathbf{Q}_{es}$. A pseudo-density is computed for each node and determines the amplitude of the perturbation such that nodes located in void area remain unaffected while nodes belonging to the structure are strongly affected and imposed to the desired electric potential.

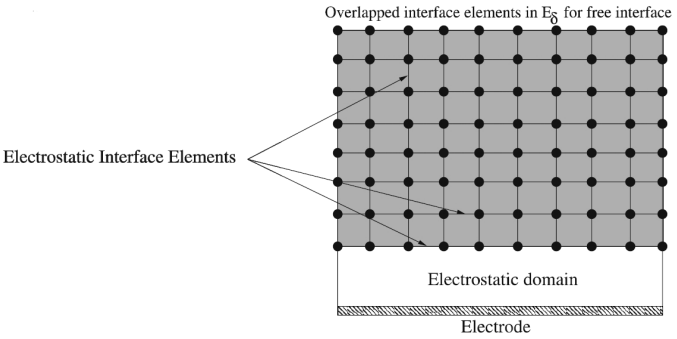


Figure 3.19: Interface element mesh [86].

Electrostatic forces are computed using interface elements. These elements are normally placed only on the boundary between void and solid. However this boundary being blurred and mobile, Raulli and Maute propose to extend the interface finite element mesh to the whole optimization domain as presented in Figure 3.19. The permittivity of the interface element is modified using a power law in order to take pseudo-density variations into account:

$$\varepsilon_j^i = \varepsilon_0 \frac{(\mu_j - \mu_{min})^{p_i}}{(1 - \mu_{min})^{p_i}}.$$

This interpolation has for consequence to allow usual calculation of the electrostatic forces on solid elements and results in zero electrostatic forces in void areas.

Application

One application of the design method presented by Raulli and Maute consists in tailoring a 2D force inverter. The problem is sketched in Figure 3.20. The compliant mechanism has to convert the electrostatic force acting downwards into an upward displacement of point C (see Figure 3.20). The fixed electrode is located at the bottom of

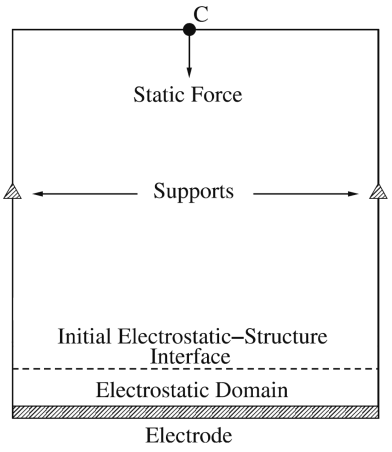


Figure 3.20: Sketch of the 2D force inverter design domain [86].

the modeling domain. The optimization domain is separated from this electrode by a purely electrostatic domain (non-designable). Supports are placed on both side of the optimization domain.

The optimization problem considered by Raulli and Maute is similar to the one presented in Section 3.5.1 for the design of electrothermal actuators:

$$\max_{\boldsymbol{\mu}} u_{out}(\boldsymbol{\mu}) = \max_{\boldsymbol{\mu}} u_C(\boldsymbol{\mu}),$$
$$s.t. \begin{cases} v(\boldsymbol{\mu}) \leqslant \overline{v}, \\ \Pi \leqslant \Pi_0, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i. \end{cases}$$

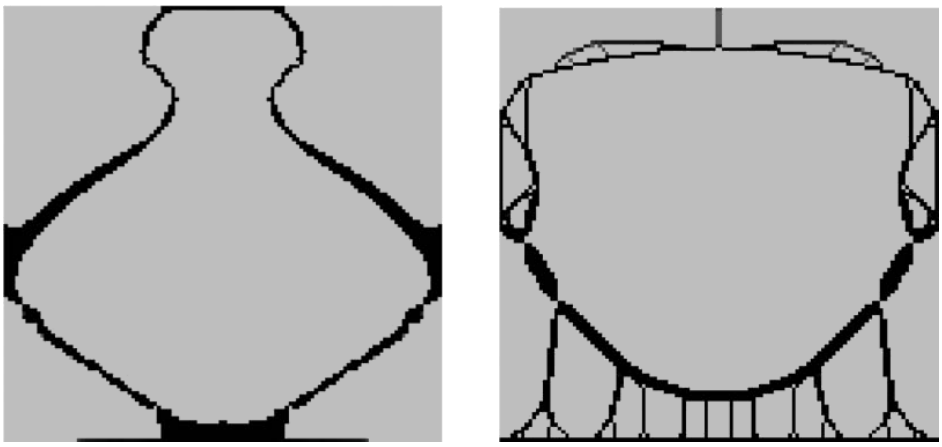
However, one additional constraint is included in the optimization problem by Raulli and Maute. This constraint prevents the strain energy Π from increasing beyond its initial value Π_0. Raulli and Maute invoke two reasons for adding this constraint:

- Firstly, they want to avoid pull-in effect. As electrostatic forces increase with the displacement of the mobile electrode (the gap decreases), the optimizer may attempt to create a very compliant structure to reach large deformation. The purpose of the strain energy constraint is to limit deformation of the structure so that it remains stable.

- Secondly, the energy constraint result in a more 0-1 material distribution as it forces the optimizer to stiffen the actuator.

Two methodologies have been tested by Raulli and Maute. The first one consists in letting the optimizer distribute material freely over the domain. Alternatively, in the second method, they imposed the lowest row of the optimization domain to be solid. As a result, the interface between electrostatic domain and optimization domain is fixed.

The results provided by these two methodologies are presented respectively in Figure 3.21(a) and Figure 3.21(b). We can notice that when allowed, the optimizer removes material from the interface between the two domains. Nevertheless, even if the mobile electrode is smaller and produces less electrostatic forces with 'free interface' design, the resulting actuator is about three times more efficient. According to the authors, in the 'free interface' design most of the material is used to stiffen the actuator. Conversely, in the 'imposed interface' design the optimizer has to use a lot of material to create a suspension structure for the imposed electrode, which weakens the actuator and reduces its efficiency.



(a) Free interface. (b) Imposed interface.

Figure 3.21: Electrostatic 2D force inverter design by Raulli and Maute [86].

Monolithic approach

As illustrated in the previous section, the use of a staggered modeling approach for topology optimization of electrostatic actuators leads to quite complex modeling and optimization procedure. However, as presented by Yoon and Sigmund [129] it is possible to develop a simpler and more consistent optimization procedure by replacing the staggered modeling by a monolithic approach. As presented in Chapter 2, monolithic approach consists in solving both physical fields simultaneously as a unified problem. In the context of topology optimization, a monolithic approach presents the advantage

to handle implicitly the boundary conditions linking the two physical fields (e.g. electrostatic forces and imposed voltage). Therefore, there is a better consistency between modeling and optimization.

The monolithic formulation proposed by Yoon and Sigmund is very similar to the one presented in Chapter 2. Similarly to Raulli and Maute, a fictitious permittivity $\tilde{\varepsilon}$ is introduced to ensure electrical conductor behavior for the structural part of the design. This fictitious permittivity takes a high value (about 10^5 times the actual value) for solid elements. To avoid non-physical modification of the electrostatic forces, fictitious permittivity is only used in the electrostatic problem equation while the actual permittivity is preserved for the electrostatic forces computations.

Similarly to Raulli and Maute, material properties are interpolated using a SIMP law. As Yoon and Sigmund distinguish the actual permittivity ε used for computing the electrostatic force from the fictitious one applied for the electrostatic problem $\tilde{\varepsilon}$, they get three interpolation equations:

$$\begin{aligned} E(\mu) &= E_{min} + \mu^p (E_s - E_{min}), \\ \varepsilon(\mu) &= \varepsilon_0 + \mu^p (\varepsilon_s - \varepsilon_0), \\ \tilde{\varepsilon}(\mu)\, \varepsilon_0 &+ \mu^p (\tilde{\varepsilon}_{max} - \varepsilon_0), \end{aligned}$$

where E_{min} is the Young Modulus affected to void areas of the domain, ε_s and ε_0 are respectively the actual dielectric permittivity of the design material and of void, and $\tilde{\varepsilon}_{max}$ is the fictitious permittivity affected to solid portions of the optimization domain. Yoon and Sigmund notice that localized instability modes may appear if the penalty parameter p appearing in both permittivity interpolations is lower than the one used for Young modulus material model. Therefore, they choose to use identical penalty for all interpolation functions. They get satisfactory results with the penalty parameter p fixed to 3.

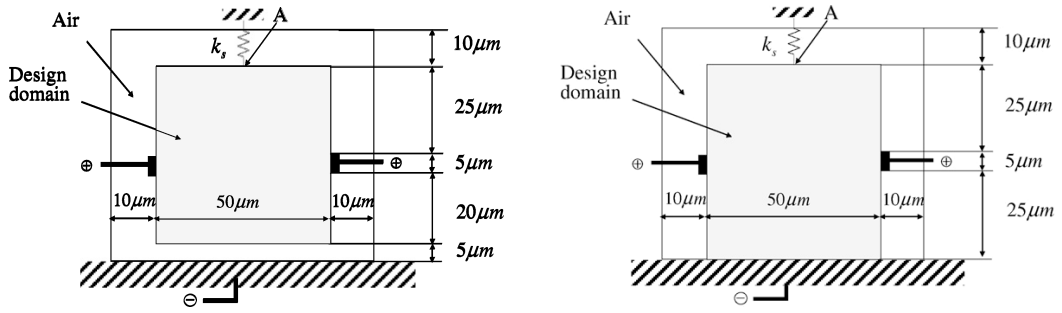
While Raulli and Maute consider output displacement, Yoon and Sigmund choose to maximize the output force of the actuator. Their optimization problem states:

$$\begin{aligned} &\max_{\boldsymbol{\mu}} f_{out}(\boldsymbol{\mu}) = \max_{\boldsymbol{\mu}} (k_s \cdot u_{out}(\boldsymbol{\mu})), \\ &s.t. \begin{cases} v(\boldsymbol{\mu}) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i, \end{cases} \end{aligned}$$

where k_s is the stiffness of the spring placed at the output port to simulate a work piece.

Yoon and Sigmund present microactuator design applications with increasing optimization problem complexity. At first, they consider that the optimization domain is separated from the fixed electrode by a non-designable air gap as illustrated in Figure 3.22(a) and they try to maximize the upward force at point A. However, they notice that the inclusion of a predefined air gap reduces strongly the freedom of the optimization problem as the location, the direction and the magnitude of electrostatic force is de facto

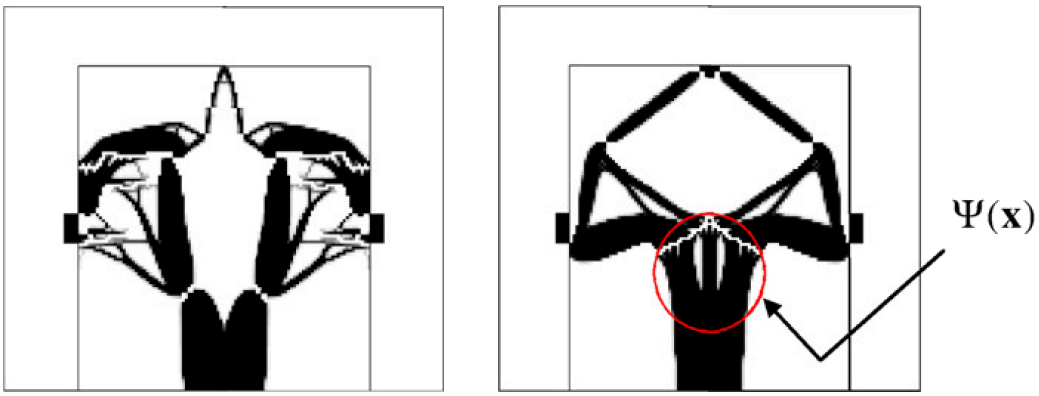
imposed. Indeed, with a predefined air gap, the optimizer should normally assign material as close as possible from the fixed electrode in order to maximize electrostatic force.



(a) With predefined air gap. (b) Without predefined air gap.

Figure 3.22: Electrostatic actuator optimization problems by Yoon and Sigmund [129].

Therefore they find more interesting to consider an optimization problem without predefined air gap as illustrated in Figure 3.22(b). The solution they obtain for this second optimization problem is presented in Figure 3.23(a). The resulting design is quite complicated and the device operating principle is not clear.

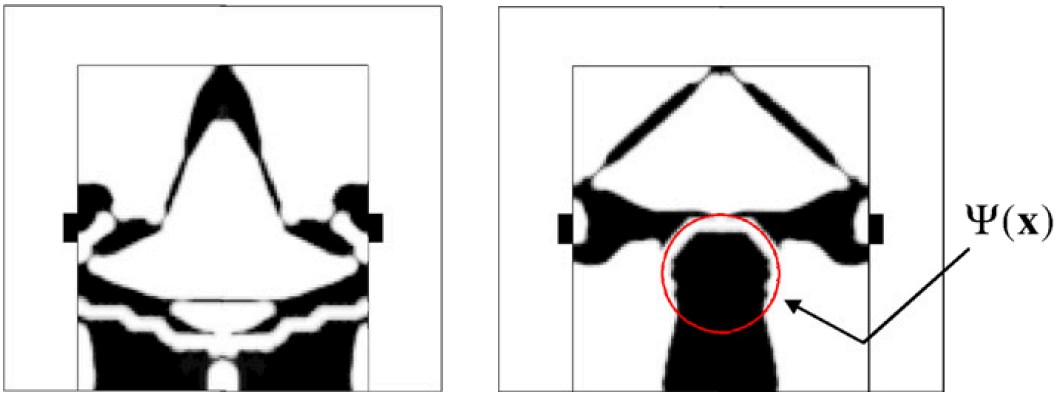


(a) Without limited actuation area. (b) With limited actuation area.

Figure 3.23: Electrostatic actuator optimization results without predefined air gap [129].

To limit the complexity of the design they choose to restrict the area of the design domain where electrostatic forces can be generated using an artificial modification of the permittivity. Figure 3.23(b) shows the design obtained when actuation is only allowed in the circled area. The device layout is simpler and the optimizer only creates one mobile electrode linked to the voltage input point and one fixed linked to the bottom of the domain. However, the gap between these two electrodes is very small and corresponds to the size of one element. This result makes sense, since the electrostatic force is inversely proportional to the square of the gap. The optimizer tends to create gaps as small as possible.

Nevertheless, the tendency to create gaps of the size of one element makes the optimization results quite mesh dependent. To circumvent this problem, Yoon and Sigmund propose to use morphological filters developed previously by Sigmund [100]. These filters are inspired from image processing techniques and allow controlling size of the features that appear in the final design. By using the *erode* morphological filter, Yoon and Sigmund are able to control the gap width between the electrodes. This is illustrated in Figure 3.24(a) for the case where electrostatic actuation can happen anywhere in the domain and in Figure 3.24(b) with a limited actuation area. In terms of output force, these designs are less efficient than the previous results from Figures 3.23(a) and 3.23(b). This can be explained by the fact that larger gaps lead to a lower electrostatic pressure.



(a) Without limited actuation area. (b) With limited actuation area.

Figure 3.24: Electrostatic actuator optimization results with gap size control [129].

The results obtained by Yoon and Sigmund are very encouraging as they show that monolithic modeling is well suited to electrostatically actuated compliant mechanisms optimization. However, they also point out two important issues they have been facing along the development of the optimization procedure. Firstly, they observe that the treated optimization problem possesses many local optima. So the final design may depend on initial settings of the optimization procedure. Secondly, they notice instabilities at mesh-scale which shows that electrostatic actuator design should include stability constraint.

Robust design with respect to manufacturing errors

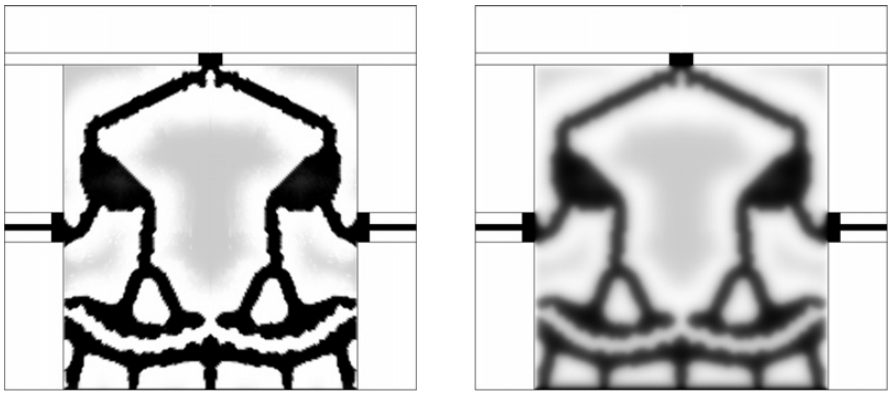
Recently, Qian and Sigmund [84] extended the approach proposed by Yoon and Sigmund [129] by including a density filtering method based on Helmholtz partial differential equation followed by a Heaviside projection. This filtering procedure allows controlling minimal length for both electrostatic gaps and structural features. Moreover, a multi-objective optimization problem is formulated by using three different thresholds for the Heaviside projection in order to improve the robustness of the final design with respect

to over- and under-etching manufacturing errors. The modeling framework is identical to the monolithic formulation proposed by Yoon and Sigmund [129].

The Helmholtz density filter is a PDE based alternative to explicit density filtering (see for instance Eq. (3.3)). As proposed by Lazarov and Sigmund [57], this alternative approach consists in computing the density field $\tilde{\mu}$ as the solution of the following equation:

$$-r^2\nabla^2\tilde{\mu} + \tilde{\mu} = x,$$

where x corresponds to the design variables field and r is a parameter similar to the filtering radius R in classic density filtering (see Eq. (3.3)) with $r = 2\sqrt{3}R$. This partial differential equation is solved using finite element method which allows parallel implementation of the filtering process. One design variable x_i is then assigned to each node of the finite element mesh in order to represent the design variable field and the solution of the finite element problem yields the filtered nodal densities $\tilde{\mu}_i$. This process is illustrated in Figure 3.25 where the design variable field is represented in Figure 3.25(a) while the resulting filtered density field $\tilde{\mu}$ is proposed in Figure 3.25(b).



(a) Design variable field x. (b) Filtered density field $\tilde{\mu}$.

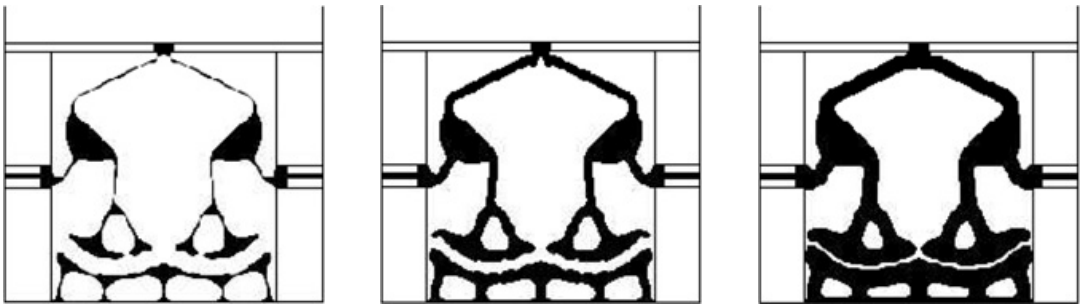
Figure 3.25: Effect of the Helmholtz PDE based density filter [84].

We can notice in Figure 3.25(b) that the filtered density field $\tilde{\mu}$ contains a lot of non-integer densities. Therefore, in order to obtain 0-1 density field, the physical density field μ is computed by applying a smoothed Heaviside projection derived from the one proposed by Sigmund [100], Guest *et al.* [43] and Xu *et al.* [124]:

$$\mu_i = \frac{\tanh(\beta\eta) + \tanh(\beta(\tilde{\mu}_i - \eta))}{\tanh(\beta\eta) + \tanh(\beta(1-\eta))}, \tag{3.7}$$

with η being the projection threshold and β determining the sharpness of the projection. At the beginning of the optimization process, β is initialized to a small value such that the projection function is smooth. As the optimization progresses, the value of β is increased in order to get a sharper projection function more similar to the actual Heaviside projection leading to crisp designs.

Using three different values for the η threshold in Eq. (3.7), one defines three realizations for the same design variables, namely $\boldsymbol{\mu}^e$ (eroded, $\eta_e \in [0.5, 1]$), $\boldsymbol{\mu}^i$ (blue-print, $\eta_i = 0.5$) and $\boldsymbol{\mu}^d$ (dilated, $\eta_d = 1 - \eta_e$). Considering the filtered density field from Figure 3.25(b) we can get the three projections presented in Figure 3.26 using a threshold $\eta_e = 0.75$. The eroded design corresponds to the case of over-ecthing while the dilated design would appear in the event of under-etching.



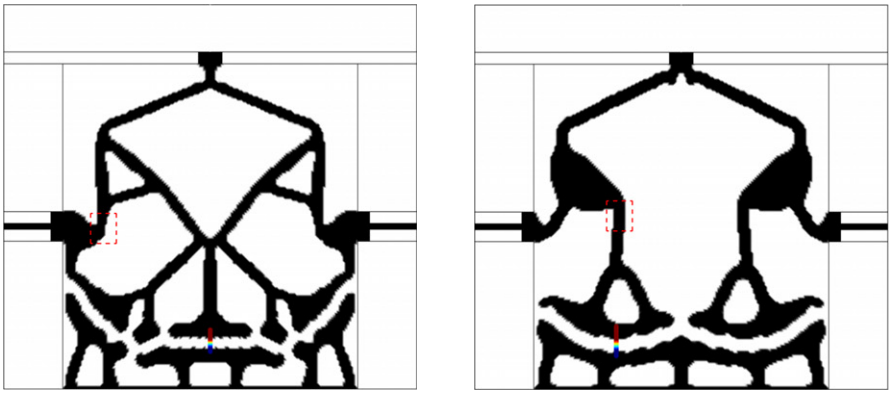
(a) Eroded $\boldsymbol{\mu}^e$, $\eta = 0.75$. (b) Blue-print $\boldsymbol{\mu}^i$, $\eta = 0.5$. (c) Dilated $\boldsymbol{\mu}^d$, $\eta = 0.25$.

Figure 3.26: Heaviside projection for three different thresholds [84].

The principle of robust design is then to include the performance of the three realizations $\boldsymbol{\mu}^e$, $\boldsymbol{\mu}^i$ and $\boldsymbol{\mu}^d$ into a multiobjective optimization problem:

$$\max_{\mathbf{x}} \min \left(f_{out}\left(\boldsymbol{\mu}^e\left(\mathbf{x}\right)\right), f_{out}\left(\boldsymbol{\mu}^i\left(\mathbf{x}\right)\right), f_{out}\left(\boldsymbol{\mu}^d\left(\mathbf{x}\right)\right)\right),$$
$$s.t. \begin{cases} v\left(\boldsymbol{\mu}^d\right) \leqslant \overline{v}, \\ x_{min} \leqslant x_i \leqslant 1 \quad \forall i. \end{cases}$$

Qian and Sigmund demonstrate the efficiency of the proposed approach on the numerical application already presented in Figure 3.22(b). This design problem is solved for two different filtering radii, $R = 6.3h$ and $R = 8.4h$ with $h = 0.3333$ µm being the element size. The resulting blue-print realizations $svec\mu^i\left(\mathbf{x}\right)$ are depicted in Figure 3.27. The optimization procedure leads to very sharp 0-1 designs. Compared to the results obtained by Yoon and Sigmund [129] in Figure 3.24(a), one can verify that the PDE based filter allows ensuring minimal length control for both electrostatic gaps and structural hinges. This leads to an improvement of the electromechanical modeling accuracy. In order to predict the minimal length imposed by the filter, Qian and Sigmund developed analytical formulas depending on η_e and on R which shows good agreement with optimization results. From the point of view of the performances, for each actuator in Figure 3.27, the output force produced by eroded, blue-print and dilated realizations are identical. This illustrates the capabilities of the method to generate robust designs toward manufacturing tolerances. Finally, like previous studies on the topic, they noticed the appearance of pull-in instability in some applications.



(a) $R = 6.3h$. (b) $R = 8.4h$.

Figure 3.27: Robust electrostatic actuator design for different filter radii [84].

3.6 Conclusion

Topology optimization has several advantages with respect to shape optimization and sizing optimization. The possibility to add or remove structural components enlarges the design space and makes topology optimization more efficient. Furthermore, topology optimization allows determining the optimal material layout without any a priori on the connectivity of the structure. Consequently, it reduces the arbitrariness in the choice of the designer.

This chapter describes two approaches to implement topology optimization. Among them, the homogenization based method is mature and the most widely applied. Indeed, firstly, initial difficulties with ill-conditioning of the optimization problem can be solved by the use of filtering techniques. Secondly, state of the art presented in this chapter shows that this method can be applied to several multiphysic problems including electrostatic microdevices optimization. Finally, homogenization based method is simpler to implement on top of an existing finite element code than the level set method which may require solving Hamilton-Jacobi equation and special numerical techniques to model the material distribution.

For these reasons, topology optimization based on homogenization method has been selected. Moreover, in this thesis, we could take advantage of the existing monolithic finite element code developed by Rochus [91]. Therefore, we could focus on the construction of a topology optimization procedure able to control pull-in voltage following the tracks opened by Yoon and Sigmund [129], Qian and Sigmund [84] and Abdalla *et al.* [1].

Chapter 4

Simplified topology optimization problem for pull-in voltage maximization

4.1 Introduction

One of the aims of the present thesis consists in developing a topology optimization procedure able to control pull-in voltage in electrostatically actuated microdevices. This chapter is dedicated to a first step in the achievement of this task; including a pull-in voltage criterion into topology optimization problem. In the scope of this chapter, the pull-in voltage function takes the place of the objective function in order to keep the optimization problem as concise and as simple as possible.

The principal contributions to topology optimization of electromechanical microsystems have been presented in Chapter 3. As shown by Raulli and Maute [86], the main difficulty of electromechanical microdevices topology optimization stems from the design dependency of electrostatic forces. Indeed, the location of these forces depends on the material layout as they are applied at the surface of the structure. Therefore, a modification of the material distribution modifies the application surface as well as the direction and intensity of the electrostatic forces. Moreover, in topology optimization context, the surface of the structure is usually not accurately defined since there is often a smooth transition between void and solid.

For these reasons, the research work has been initiated on basis of a simplified optimization problem in which the location of the electrostatic forces is fixed. As described in the present chapter, this simplified version of the optimization problem allows separating the difficulties. On basis of the simplified problem, a pull-in voltage optimization procedure is developed while keeping away some of the issues of the full problem. Indeed, in later research, Yoon and Sigmund [129] notice that local instability problems may arise when considering the coupled optimization problem even though they used a

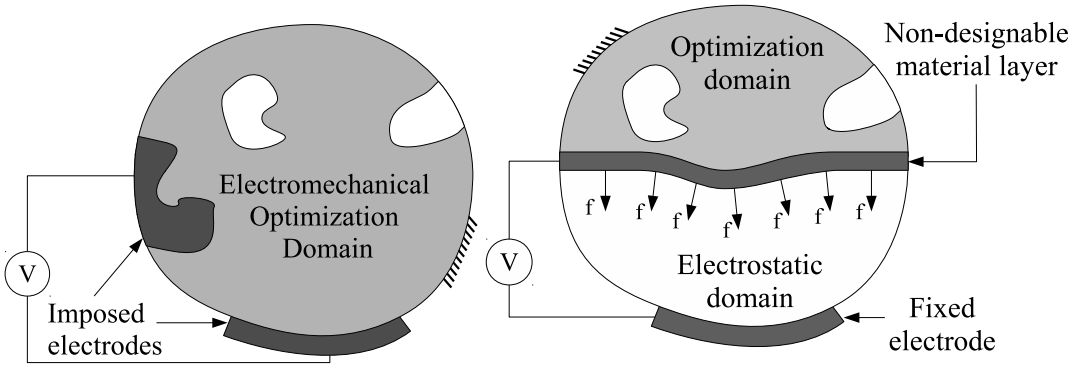
monolithic modeling. The approach proposed in this chapter corresponds to a first step in order to establish an optimization procedure for the full problem.

After the description of the optimization procedure and numerical applications, this chapter also considers the addition of a manufacturing constraint to the optimization problem. The goal of this constraint is to make the final design more compliant with manufacturing techniques used for MEMS manufacturing.

4.2 Simplified optimization problem

4.2.1 Optimization problem simplification

The simplification of the optimization problem considered in this chapter essentially consists in assumptions on the topology of the modeling domain and on the place that the optimization domain takes in it. Without these assumptions, the modeling domain associated with topology optimization of electrostatically actuated microdevices takes the general form represented in Figure 4.1(a). In this case, the optimization domain (in light gray) covers most of the modeling domain excepting imposed void (in white) and solid (in dark gray) areas. The mechanical and the electrostatic physical fields interact over the optimization domain which can be described as multiphysic. Consequently, the modifications of the material layout have a direct influence on both physical fields. As previously stated, electrostatic forces are design dependent and the optimizer is free to adapt the electrostatic forces location.



(a) General optimization problem. (b) Simplified optimization problem.

Figure 4.1: Schematic view of the considered optimization problem.

In order to simplify the optimization problem and to fix the electrostatic forces position, the modeling domain is split in two parts as presented in Figure 4.1(b). The optimization domain is restricted to one part of the modeling domain and is insulated from electrostatic effects by a non-designable material layer. The other part of the modeling domain is non-designable and imposed to be void. Consequently, the optimization domain can be considered as purely mechanical and electrostatic forces application surface

is fixed on the non-designable material layer. In other words, the optimization problem consists in designing an optimal suspension for the mobile electrode. However, the optimization problem is still multiphysic and strongly nonlinear since the interaction between mechanical and electric phenomena remains.

To develop and test the possibility to control pull-in voltage using topology optimization, we consider the problem of maximizing pull-in voltage of an electromechanical device with a bound over the available volume of material. This can be mathematically stated as follow,

$$\max_{\boldsymbol{\mu}} V_{pi}\left(\boldsymbol{\mu}\right),$$
$$s.t. \begin{cases} v\left(\boldsymbol{\mu}\right) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i, \end{cases}$$

where $\boldsymbol{\mu}$ represents the vector of pseudo-densities, $\overline{v}$ the upper bound on available material volume, $v\left(\boldsymbol{\mu}\right)$ the structure volume, μ_{min} the lower bound on the pseudo-densities. The design material is an elastic-linear material under small strains assumption.

The use of pull-in voltage as objective function is only for development and testing purposes. This choice simply allows us keeping the optimization problem as simple as possible by limiting the number of design functions. Moreover, later transforming an objective function into a design constraint is straightforward into our mathematical programming framework. Therefore, the method developed here is easily transposable to other optimization problems where pull-in voltage function is assigned as a design constraint (see Chapter 7).

Additionally, in order to focus on the development of the pull-in design function, we choose to rely on classical sensitivity filtering (see Eq. (3.2)) or density filtering (see Eq. (3.3)) for the regularization of the optimization problem along this thesis. Consequently, the use of more advanced filtering schemes as the one proposed by Qian and Sigmund [84] is not considered in the present work.

4.2.2 Sensitivity analysis

Considering the optimization problem stated above, it is possible to obtain an expression of the pull-in voltage sensitivities as a function of the pull-in conditions. The sensitivities equation can be evaluated starting from the equilibrium equation by adapting the reasoning proposed by Abdalla *et al.* [1] to the strongly coupled formulation. Using the notation from Chapter 2, the equilibrium equation writes:

$$\mathbf{r}\left(\mathbf{q}, V\right) = \mathbf{f}_{int} - \mathbf{f}_{ext} = 0,$$

where $\mathbf{r}$ stands for the residual force vector, $\mathbf{q}$ are the generalized displacements and V is the applied electric potential. The equilibrium equation is then derived with respect to the design variable μ_i:

$$\frac{\partial \mathbf{r}}{\partial \mu_i} + \frac{\partial \mathbf{r}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial \mu_i} + \frac{\partial \mathbf{r}}{\partial V} \frac{\partial V}{\partial \mu_i} = \mathbf{0}.$$

Notice that the perturbation of a design variable will not solely modify the pull-in voltage but also the deformation state at pull-in point as illustrated in Figure 4.2. Thus,

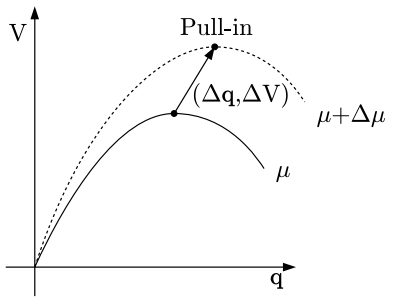


Figure 4.2: Evolution of the pull-in curve resulting from a perturbation of x_i.

both derivatives of V and $\mathbf{q}$ with respect to μ_i have to be considered. In the scope of this thesis, no external forces are considered so that $\mathbf{f}_{ext} = \mathbf{0}$ and that $\mathbf{r} = \mathbf{f}_{int}$. Therefore, as in Section 2.5.3, the derivative of the residual forces with respect to generalized displacements is by definition equal to the tangent stiffness matrix:

$$\frac{\partial \mathbf{r}}{\partial \mu_i} + \mathbf{K}_t \frac{\partial \mathbf{q}}{\partial \mu_i} + \frac{\partial \mathbf{r}}{\partial V}\frac{\partial V}{\partial \mu_i} = \mathbf{0}.$$

Then, using equation (2.13) the derivative of the residual forces becomes:

$$\frac{\partial \mathbf{r}}{\partial \mu_i} + \mathbf{K}_t \frac{\partial \mathbf{q}}{\partial \mu_i} + \mathbf{K}_t^{f,i}\frac{\partial \mathbf{q}^i}{\partial V}\frac{\partial V}{\partial \mu_i} = \mathbf{0}. \tag{4.1}$$

Next, let's pre-multiply this equation by the first eigenvector $\mathbf{p}$ of $\mathbf{K}_t$. Since this matrix is singular at pull-in point we have $\mathbf{K}_t\mathbf{p} = \mathbf{0}$. In addition, the eigenvector is normalized to have:

$$\mathbf{p}^T\mathbf{K}_t^{f,i}\frac{\partial \mathbf{q}^i}{\partial V} = -1.$$

Under this condition, the multiplication of Eq. (4.1) by $\mathbf{p}$ gives:

$$\mathbf{p}^T\frac{\partial \mathbf{r}}{\partial \mu_i} - \frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{0}.$$

This leads us to the expression of pull-in voltage sensitivities:

$$\frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}^T\frac{\partial \mathbf{f}_{int}}{\partial \mu_i}. \tag{4.2}$$

Notice that the assumption of a purely mechanical design domain has not been used up to now. Therefore, the last sensitivity equation is valid even in the general case (i.e. multiphysic optimization domain).

In the particular case where the optimization domain is purely mechanical, the variation of $\mathbf{r}$ resulting from a density perturbation comes solely from the mechanical contribution

to the internal forces $\mathbf{f}_{int}$. As linear mechanics are considered, the mechanical contribution to internal forces is equal to the product of stiffness matrix with generalized displacements. If $\mathbf{K}$ denotes the linear stiffness matrix, this gives:

$$\frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}^T \frac{\partial \mathbf{K}}{\partial \mu_i} \mathbf{q}. \tag{4.3}$$

Moreover, the partial derivative of the electrostatic part of the stiffness matrix $\mathbf{K}$ is null since design variables do not affect directly the electrostatic part of the domain. Therefore, we can restrict Eq. (4.3) to the mechanical part of matrix $\mathbf{K}$ and finally write:

$$\frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}_u^T \frac{\partial \mathbf{K}_{uu}^0}{\partial \mu_i} \mathbf{u}, \tag{4.4}$$

with $\mathbf{p}_u$ representing the mechanical part of the eigenmode $\mathbf{p}$ and $\mathbf{K}_{uu}^0$ being the mechanical linear stiffness matrix as defined in Section 2.4.2.

The derivative of the mechanical stiffness matrix is simple to evaluate. Indeed, matrix $\mathbf{K}_{uu}^0$ only depends on the pseudo-density μ_i through the Young modulus of the corresponding element which is given as a function of μ_i by the material model. Additionally, the tangent stiffness matrix eigenmode $\mathbf{p}$ and the generalized displacements $\mathbf{q}$ need to be evaluated at pull-in point. Consequently, they require the use of path following procedure up to pull-in point. However, as they are common to all variables, the sensitivities of pull-in voltage with respect to every variables requires only one pull-in search.

4.2.3 Optimization procedure

All the elements of the optimization procedure have now been presented. Figure 4.3 presents the organization of these different steps that shape the optimization program. The optimization procedure is based on the classic scheme of structural optimization. First, the design analysis is performed using the normal flow algorithm described in Section 2.5.1. The analysis aims at determining the pull-in point of the current design. It computes the generalized displacements vector $\mathbf{q}$ and the eigenmode of the tangent stiffness matrix $\mathbf{p}$ at pull-in point. Then, sensitivities are evaluated using results of the analysis and Eq. (4.4). Next, on basis of these sensitivities, *Conlin* optimizer [41] is used to yield a design update $\Delta\boldsymbol{\mu}$. The stopping criterion is usually based on the design update (mean or max variation) but can also take into account a maximum number of iterations. If the stopping criterion is not satisfied, the optimization loop continues with the updated design otherwise the optimization is stopped.

4.3 Numerical applications

The efficiency and the stability of the optimization method developed here above are illustrated on the basis of two numerical applications. The results provided by the pull-in optimization procedure are analyzed and in the case of the second application they

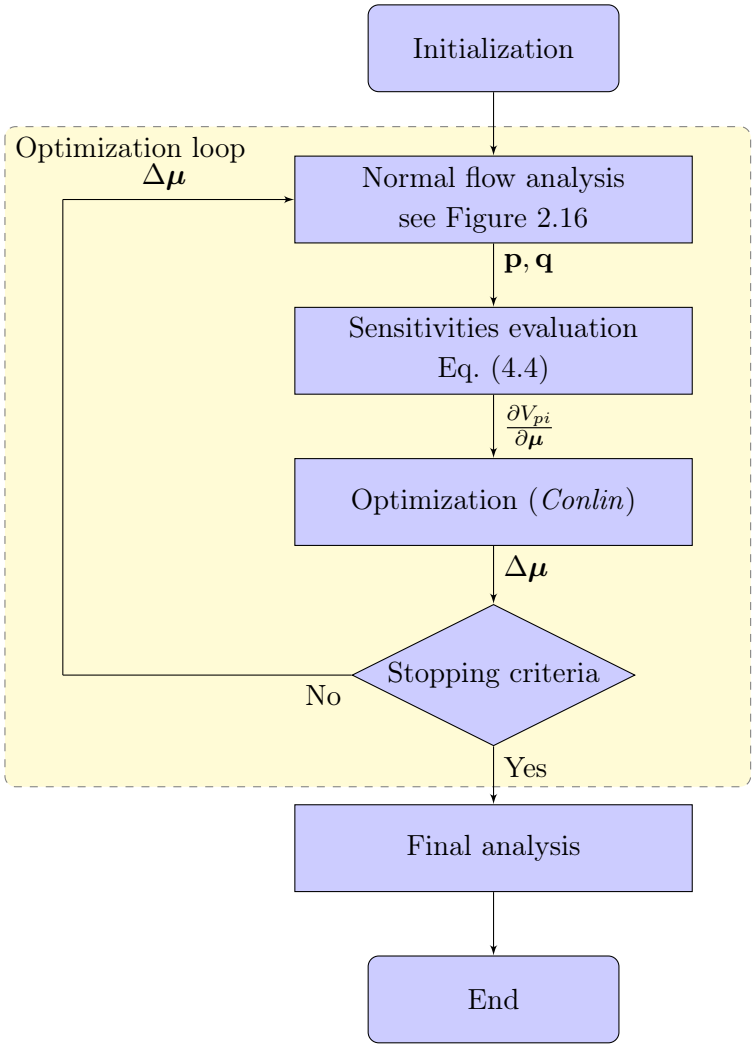


Figure 4.3: Optimization procedure flowchart.

are compared to the linear compliance optimization results. Moreover, quality of the sensitivities provided by the semi-analytic procedure is assessed with respect to finite difference.

4.3.1 Clamped-clamped microbeam suspension optimization

The first application consists in designing a suspension structure for a clamped-clamped microbeam that maximizes pull-in voltage. The optimization problem as well as the boundary conditions are sketched in Figure 4.4. The clamped-clamped beam is represented in black. It separates the optimization domain from the electrostatic domain. A voltage difference is applied between the fixed electrode and the clamped-clamped beam. The optimization domain is placed over the clamped-clamped beam. As the problem is symmetrical, only one half is modeled. The modeling domain is meshed with 100×25 finite elements which results in 2000 design variables over the optimization domain.

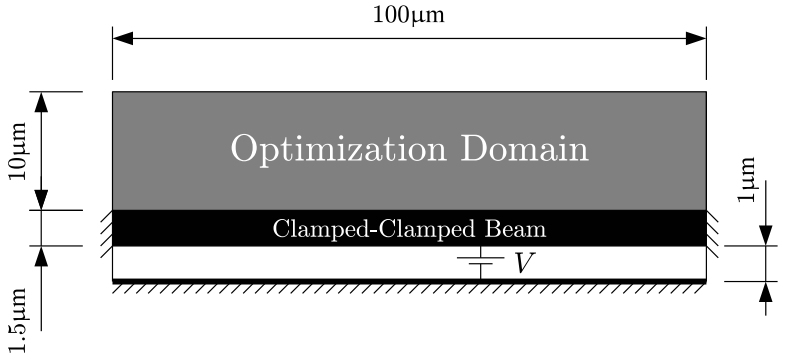


Figure 4.4: Clamped-clamped beam optimization problem and boundary conditions.

The design material is isotropic quartz whose mechanical properties are listed in Table 4.1. This material is also used for the imposed microbeam. The available volume of design material is limited to 40% of the design domain volume. The gap between the mobile electrode and the fixed electrode is filled with air.

Young Modulus	86790 MPa
Poisson ratio	0.17

Table 4.1: Design material physical properties.

The SIMP model is used to compute material properties of intermediate density elements. Parameter p is set to 3 for the present example. Finally, sensitivity filtering is applied to regularize the optimization problem with a filtering radius equal to 1.5 times the finite element size (0.75 µm).

The optimization process is stopped when the maximum design parameter variation drops below 0.01. This stopping criterion is reached after 74 iterations. The final

topology is presented in Figure 4.5. The suspension is very similar to a bridge structure with its double arch linked to the imposed beam by oblique bars.

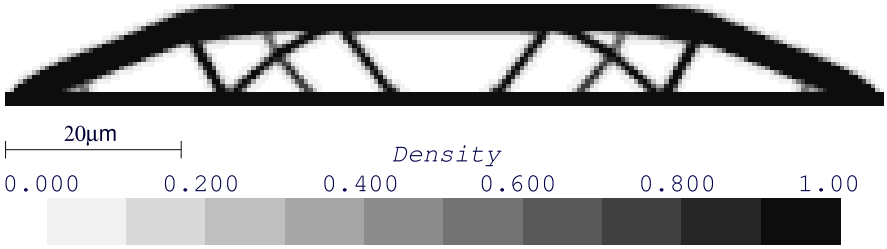


Figure 4.5: Clamped-clamped beam final topology.

The pull-in voltage of the optimized structure is of 678 V which corresponds to an increase of 140% with respect to the initial uniform distribution design. The history of pull-in voltage across optimization iterations is presented in Figure 4.6. As one can see, the curve presented in the figure is smooth and free of oscillations. Pull-in voltage increases strongly during first iterations and has almost converged to its final value after 30 iterations even though more than 40 additional iterations are needed to reach stopping criterion.

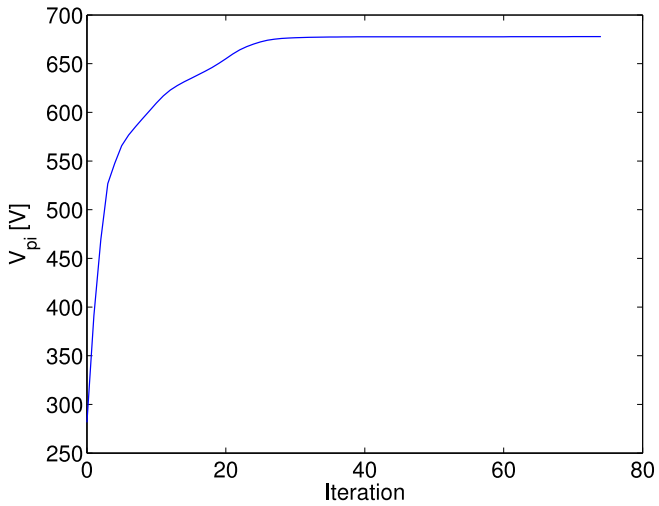


Figure 4.6: Clamped-clamped beam optimization: history of pull-in voltage.

Sensitivity verification

The validity of the pull-in voltage semi-analytic sensitivity computation has been verified by comparing its results with the sensitivities obtained using finite differences. This comparison has been performed on basis of the last optimization problem (see Figure 4.4) with a random material distribution. As a result, the relative difference between finite difference and semi-analytic sensitivities is lower than 1% for each design

variable. This confirms the reliability of the semi-analytic pull-in voltage sensitivity expression evaluation proposed in Section 4.2.2 and the validity of the complete analysis procedure as well.

4.3.2 Microbeam suspension optimization with off-center electrode

The second test case is presented in Figure 4.7. Unlike the last example the ground electrode is limited to the left half of the optimization domain. As a result there are almost no electrostatic forces acting on the right half of the microbeam and the problem is asymmetrical. Moreover, fixations are placed on both sides of the optimization domain so that the optimizer can take advantage of additional fixation points to improve the structure.

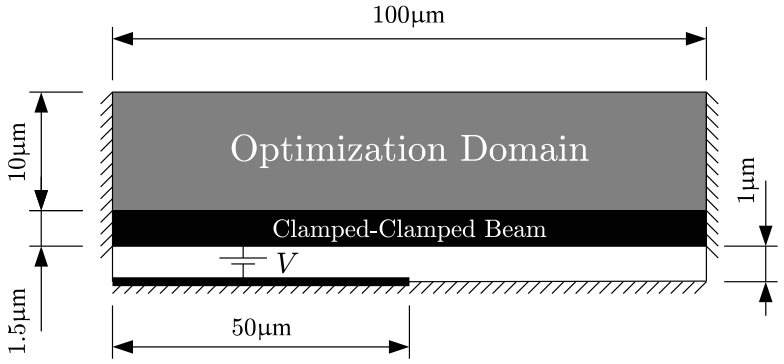


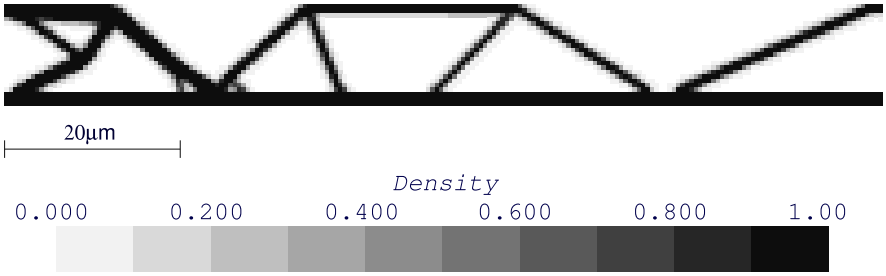
Figure 4.7: Microbeam with off-center electrode optimization problem and boundary conditions.

The design domain is meshed with 200 times 20 quadrangular elements which results in 4000 design variables; that is two times more than in the previous application since symmetry cannot be used here. Mesh dependency and checkerboard pattern are avoided through the use of sensitivity filtering with radius equal to 1.5 times the finite element size.

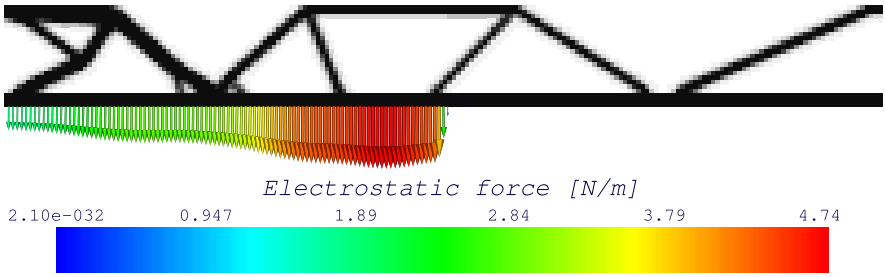
Similarly to previous application, a volume constraint bounds the available volume of design material to 20% of the design domain volume. The design material is the same isotropic quartz as in the last application and SIMP law is used to model intermediate density element properties. However, conversely to the last example, setting penalization parameter p to 3 is not sufficient to remove intermediate densities from the final design. Therefore, p has been gradually increased from 3 to 4 from first iteration to iteration 100 and remains constant afterward.

The optimized design presented in Figure 4.8(a) is obtained after 337 iterations. The resulting structure presents a bridge shaped structure similarly to last application. However, thanks to the fixations provided on the sides of the optimization domain, the optimizer shortens the span of the arch. On both sides of the design domain, supporting structures are anchored at the upper corners while the bridge structure only covers the

central part of the optimization domain. Moreover, as expected, the suspension structure is asymmetrical because of the off-center fixed electrode. The electrostatic pressure distribution at pull-in point is presented in Figure 4.8(b). We can see that the structure has been tailored to the loading. The left part of the design presents a more complex truss with thicker members than the right side. Under the central arch structure, the two members linking the arch to the imposed beam are clearly asymmetrical and aim at supporting the electrostatic load. Finally, the effects of the structure deformation on the electrostatic forced magnitude are clearly visible in Figure 4.8(b). Larger deflection at the center of the beam leads to a smaller effective gap with respect to the fixed electrode. By consequence, the magnitude of the electrostatic forces is higher at the center of the beam.



(a) Final topology.



(b) Electrostatic forces distribution.

Figure 4.8: Microbeam suspension with off-center electrode optimization results.

4.3.3 Comparison with linear compliance optimization

In the two last applications, maximization of pull-in voltage results in a bridge structure design over the microbeam. The physical interpretation of this behavior is simple. The optimizer tends to maximize the stiffness of the structure with respect to the electrostatic pressure. For instance, if we consider the simplified electrostatic actuator from Section 2.3 and in particular the expression of its pull-in voltage (Eq. (2.3)), it is clear that increasing the suspension spring stiffness leads to a greater pull-in voltage value.

Maximizing the stiffness of a linear structure is one of the most studied problems in the field of structural optimization (see [19, Chap. 1]). It is generally formulated as the

minimization of the compliance (i.e. deformation energy) of the structure submitted to a constant load over the optimization problem:

$$\min_{\boldsymbol{\mu}} c\left(\boldsymbol{\mu}\right) = \mathbf{f}_m^T \mathbf{u},$$
$$s.t. \begin{cases} v\left(\boldsymbol{\mu}\right) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i. \end{cases}$$

Where $\mathbf{u}$ denotes the mechanical displacements that result from the load case $\mathbf{f}_m$. The displacement and the applied forces are linked by the linear mechanical stiffness $\mathbf{K}_{uu}^0$:

$$\mathbf{K}_{uu}^0 \mathbf{u} = \mathbf{f}_m.$$

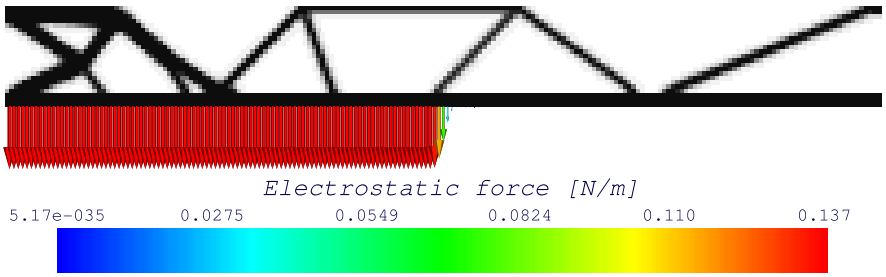
Consequently, linear compliance optimization is relatively simple and fast since it only requires linear analysis of the structure. Therefore, it is interesting to compare the last results of pull-in voltage optimization with the one provided by a simple linear compliance minimization procedure.

Prior to perform compliance optimization the load case $\mathbf{f}_m$ has to be defined. With this aim in view, two strategies have been implemented and tested. The first one is the simplest, it consists in computing the electrostatic pressure acting on the beam at rest position and using it as load case that we note $\mathbf{f}_{m,0}$. This procedure has the advantage of only requiring a linear electrostatic analysis before starting the optimization and requires only linear computations. However the resulting load case may be quite different from the one that was actually considered with pull-in optimization as the effects of the structural deformation on the electrostatic forces are neglected. The second one addresses this limitation by resorting to a nonlinear pull-in point search of the initial structure. The forces at pull-in point (noted $\mathbf{f}_{m,pi}$) are then used as constant load case throughout compliance optimization.

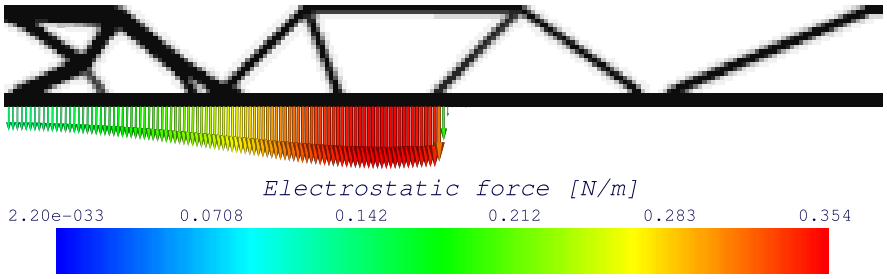
To compare compliance optimization to pull-in optimization, we choose to apply it to the microbeam with off-center electrode optimization problem (see Section 4.3.2 and Figure 4.7). The results provided by both strategies are presented in Figure 4.9(a) and Figure 4.9(b) as well as the force distribution used to formulate the optimization problem. To help results comparison Figure 4.9(c) presents the results of pull-in voltage optimization (it is identical to Figure 4.8(b)).

Figures 4.9(a) and 4.9(b) also present the load case used for compliance optimization. The difference between the two approaches is clearly visible. The electrostatic force distribution is almost uniform over the area covered by the ground electrode when computed on the undeformed configuration (Figure 4.9(a)). Conversely, pull-in deformation of the initial design leads to greater electrostatic forces on the central part of the microbeam (Figure 4.9(b)) but requires a nonlinear analysis. Nevertheless, the second force distribution is of course closer to the final force distribution that results from pull-in voltage optimization.

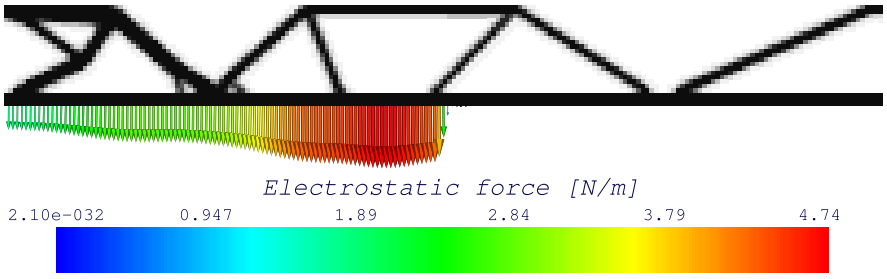
With respect to pull-in voltage optimization both compliance optimization approaches lead to slightly different topologies. The most obvious difference is the additional link



(a) Compliance optimization using rest position electrostatic pressure ($\mathbf{f}_{m,0}$).



(b) Compliance optimization using initial pull-in state electrostatic pressure ($\mathbf{f}_{m,pi}$).



(c) Pull-in voltage optimization.

Figure 4.9: Comparison of pull-in voltage optimization with compliance optimization results.

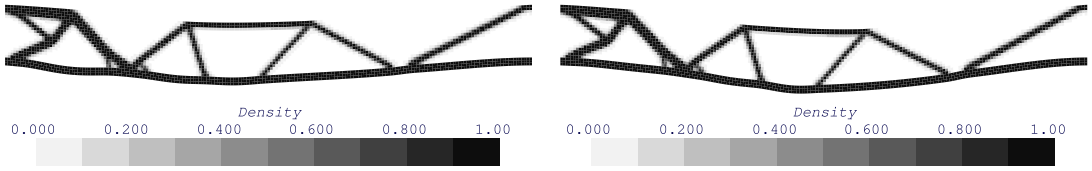
in the left part of the structure. A closer observation of the three material layouts shows that compliance optimization results in thicker members in the left part of the structure. This last effect is less marked for the second compliance optimization strategy.

These differences can be at least partly attributed to the modifications of electrostatic forces during pull-in optimization. Indeed, while the loading remains constant with both compliance optimization approaches, pull-in optimization procedure takes into account the modifications of electrostatic forces distribution caused by design updates. This difference between initial and final electrostatic pressure is visible by comparing Figures 4.9(b) and 4.9(c).

Another part of the explanation can be found by comparing the sensitivities of the two objective functions. The sensitivity to design perturbation of the compliance function (see [19, Chap. 1]) and pull-in function (see Eq. (4.4)) are given by:

$$\frac{dc}{d\mu_i} = -\mathbf{u}^T \frac{\partial \mathbf{K}^0_{uu}}{\partial \mu_i}\mathbf{u} \quad \text{and} \quad \frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}_u^T \frac{\partial \mathbf{K}^0_{uu}}{\partial \mu_i}\mathbf{u}.$$

As one can see, the two expressions are very similar. Provided that the displacement fields are equal (which implies that the loading is correctly evaluated for the compliance method), the last two factors are identical. However, the first factor, the eigenmode $\mathbf{p}$ of the tangent stiffness, must be kept in mind. Indeed, this mode corresponds to the instability mode and though it is generally similar to the mechanical displacements they are not strictly equal. This is illustrated in Figure 4.10 which shows side by side the displacements at pull-in point and the associated first eigenmode of the tangent stiffness matrix.



(a) Deformation. (b) Tangent stiffness matrix first mode.

Figure 4.10: Comparison of the displacements and the first eigenmode shape at pull-in.

The differences in the structures provided by compliance optimization have also some consequences on the final performance of the design. The efficiency of each optimization strategy have been computed in terms of the value of the three corresponding objective function. These results are gathered in Table 4.2 where each line corresponds to a design strategy and each column to an objective function.

The best value for each function is underlined in Table 4.2. Obviously, each of these 'best values' is provided by the optimization strategy based on the corresponding function; i.e. they are located on the diagonal of the table. Pull-in voltage values follow the observations made previously on the final designs. Using the initial pull-in state to compute electrostatic forces ($\mathbf{f}_{m,pi}$) leads to a greater final pull-in voltage than when

Strategy \ Final value	Compl. $\mathbf{f}_{m,0}$ [J/m]	Compl. $\mathbf{f}_{m,pi}$ [J/m]	V_{pi} [V]
Compliance $\mathbf{f}_{m,0}$	$\underline{5.33 \cdot 10^{-8}}$	$2.63 \cdot 10^{-7}$	855.9
Compliance $\mathbf{f}_{m,pi}$	$5.44 \cdot 10^{-8}$	$\underline{2.59 \cdot 10^{-7}}$	886.2
Pull-in voltage	$6.07 \cdot 10^{-8}$	$2.64 \cdot 10^{-7}$	$\underline{906.4}$

Table 4.2: Final performance according to design strategy.

using rest position electrostatic forces ($\mathbf{f}_{m,0}$). However both formulation leads to a lower pull-in voltage than pull-in voltage optimization, 5.5% with $\mathbf{f}_{m,0}$ and 2.2% with $\mathbf{f}_{m,pi}$. Conversely, the design obtained with pull-in voltage optimization is less efficient in terms of compliance when it is submitted to $\mathbf{f}_{m,0}$ and $\mathbf{f}_{m,pi}$.

However, in terms of computational time, both compliance optimization strategies have a great advantage over pull-in optimization. This can be verified in Table 4.3 that gives the total CPU time, the number of iterations and the average time per iteration for each method. As we can see, compliance optimization is about 40 times faster than pull-in optimization if time per iteration is considered and more than 50 faster regarding total time. This difference is of course due to the nonlinear analysis that requires more elaborated computations.

Strategy	Total time [s]	Nbr. iter.	Time per iter. [s]
Compliance $\mathbf{f}_{m,0}$	98.1	279	0.35
Compliance $\mathbf{f}_{m,pi}$	105.6	270	0.39
Pull-in voltage	5149.5	337	15.3

Table 4.3: Computational time statistics according to design strategy.

In the context of the simplified optimization problem, we have seen that minimum compliance designs where similar to the one obtained with the maximum pull-in voltage procedure and were less expensive computationally. Nevertheless, small differences in the final structure lead to higher final pull-in voltage when using pull-in voltage optimization. Therefore, even in the scope of the simplified optimization problem both function are not strictly equivalent as it is confirmed by the comparison of their sensitivities.

Moreover, if the assumptions characterizing the simplified optimization problem were suppressed, the difference between the two approaches is expected to be larger. Indeed, the two compliance methods rely on the capability of computing a good approximation of the final electrostatic pressure which is not possible in general if the point of application of the electrostatic forces is allowed to move during optimization as it is the case in the general pull-in voltage optimization problem.

4.4 Manufacturing constraint

The bridge shaped results obtained in the last two applications are common in topology optimization. However, these layouts contain several closed cavities and therefore are not well suited to MEMS manufacturing techniques. As described by Madou [68], most MEMS manufacturing techniques rely on thin material layer deposition and etching techniques. These methods allow building micromechanical devices on top of the substrate as well as introducing gaps between layers using sacrificial material layer. Nevertheless, the achievable out-of-plane complexity is very limited so that manufacturing of the optimal microbridges perpendicularly to the substrate plane is not conceivable. Conversely, in-plane capabilities are much better and building such microbridge structure parallel to the substrate is possible.

In order to enlarge the design manufacturability and to be able to build out-of-plane microbridges, the main difficulty lies in the closed cavities that are present in the final topology. Indeed, the practical creation of such complex cavities would require an unrealistic number of sacrificial layers. In order to avoid those closed cavities in the final design, we propose to introduce a manufacturing constraint inspired from topology optimization of casting parts developed by Zhou *et al.* [131, 133]. The constraint proposed by Zhou *et al.* imposes that all cavities of the part are oriented in the same direction which simplifies greatly the casting process. Mathematically, this is achieved by imposing that pseudo-densities decrease monotonously in the corresponding direction.

In MEMS context, cavities oriented in the upward direction are easier to produce using material deposition and/or etching processes. Therefore, the MEMS manufacturing constraint is simply expressed by imposing decreasing densities when going upward in each column of the finite element mesh as illustrated in Figure 4.11. Considering the

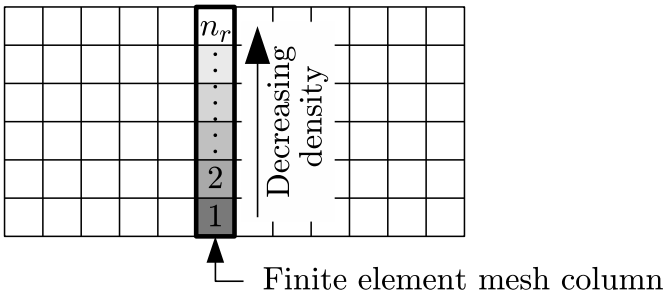


Figure 4.11: Manufacturing constraint.

design variable numbering from Figure 4.11 where n_r is equal to the number of rows of the design domain mesh, the manufacturing constraint can be expressed as follows:

$$\begin{cases} 1 & \geqslant & \mu_1, & \\ \mu_{i-1} & \geqslant & \mu_i & \text{if} \quad 2 \leqslant i \leqslant n_r, \\ \mu_{n_r} & \geqslant & \mu_{min}. & \end{cases} \tag{4.5}$$

By imposing constraints Eq. (4.5) for every column of the finite element mesh, we prevent the optimizer from closing an upward opened hole. Actually, only the second line introduces new design constraints as the first and the last one corresponds to the absolute bounds of every design variables which are already included in the optimization problem.

4.4.1 Constraint formulation

Before considering the possible formulations of the constraint, let us make a preliminary remark about linear constraints. For instance, in Eq. (4.5), all constraints are linear. Nevertheless, form the point of view of the optimizer, we can distinguish two different types of constraints. The second line is a general linear constraint while the first and the third lines correspond to so-called side constraints as they only involve one design variable that is bounded by a constant value. Unlike general linear constraints, side constraints are treated in a special way by *Conlin* or MMA dual optimizer. Thanks to this dedicated treatment described in [41], *Conlin* is able to take side constraints into account without adding dual variables. Instead, second order discontinuity planes are introduced in the dual function. Therefore side constraints are easier to handle than general linear constraints.

The introduction of the manufacturing constraint into the optimization problem results in $n_r - 1$ general linear constraints per column of the finite element mesh. In total, this gives a number of general constraints approximately equal to the number of design variables. Such a number of general constraints reduces strongly the efficiency of the dual optimizer of *Conlin*. Indeed, the advantage of treating the problem in the dual space is lost since the number of general (active) constraints is of the same magnitude as the number of (primal) design variables. Even though *Conlin* solver is generally able to carry out the optimization process, the time required to compute design update may become very prohibitive. Therefore, different possibilities have been tested in order to treat efficiently the optimization problem. This section describes the investigated formulations as well as their advantages and disadvantages.

Optimization problem reformulation

The first option to limit the number of constraints consists in finding another design parameterization such that manufacturing constraints are implicitly satisfied. Rather than parameterizing the design directly in terms of element pseudo-density and imposing Eq. (4.5), a more natural way consists in defining new design variables δ_i as the pseudo-density gap between two vertically adjacent elements. Using numbering convention from Figure 4.11, this new design parameterization corresponds to the following variable substitution:

$$\left.\begin{array}{l} \delta_1 = 1 - \mu_1 \\ \delta_i = \mu_{i-1} - \mu_i \quad \text{if} \quad 2 \leqslant i \leqslant n_r \end{array}\right\} \Leftrightarrow \mu_i = 1 - \sum_{j=1}^{i} \delta j. \tag{4.6}$$

Then, by imposing that $\delta_i \geqslant 0$, the design implicitly satisfies the first two lines of Eq. (4.5). To verify the third line ($\mu_{n_r} \geqslant \mu_{min}$), a linear constraint must be added to ensure that the sum of all δ_i does not exceed the range of variation of μ:

$$\sum_{i=1}^{n_r} \delta_i \leqslant 1 - \mu_{min}.$$

This new design parameterization seems very interesting since it reduces drastically the number of general linear constraint from $n_r - 1$ to one per column. However, in practice the resulting optimization procedure showed poor convergence properties because large modifications of the material distribution can occur within one iteration. These large design modifications are problematic since they involve significant structural response modifications that invalidate the design function approximations used by the optimizer. When working with the classical topology optimization formulation, the usual solution to this problem consists in defining move limits for all the variables of the optimization problem:

$$\underline{\mu_i^k} \leqslant \mu_i \leqslant \overline{\mu_i^k}, \tag{4.7}$$

where $\underline{\mu_i^k}$ and $\overline{\mu_i^k}$ are two bounds computed at every iteration. This strategy related to trust region methods prevents the optimizer from moving too far away from the point where the design functions were approximated. Consequently they limit the modification of the structural responses and allow ensuring the validity of the design functions approximations built by the optimizer. As move limits are side constraints, applying them on the design variables has little impact on computational load in a classical topology optimization problem while it can stabilize greatly the optimization process.

Nevertheless, in the present case defining move limits on δ_i variables is not relevant. Indeed, even if the δ_i's undergo small variations, large design modifications can occur because of the sum operator in Eq. (4.6) and involve important perturbation of the structural response. Therefore, defining side constraint on the new design variables δ_i is not a good strategy in order to impose a trust region.

Consequently, to ensure efficient trust region, the side constraints should be applied on the original design variables as in Eq. (4.7). However, if μ_i is replaced by its expression as a function of the δ_i variables Eq. (4.6), this leads us to the following linear constraints:

$$\underline{\mu_i^k} \leqslant 1 - \sum_{j=1}^{i} \delta j \leqslant \overline{\mu_i^k}.$$

All in all this makes $2n_r - 2$ additional linear constraints since upper bound (resp. lower bound) of μ_1 (resp. μ_{n_r}) are already enforced. In the end, the total number of linear constraints is higher than in the original problem and the new design parameterization does not solve the issue.

Side constraint approximation

To reduce the computational load, another possibility consists in approximating the manufacturing constraints by side constraints, namely:

$$\underline{\mu_i^k} \leqslant \mu_i^k \leqslant \overline{\mu_i^k},$$

where the lower bound $\underline{\mu_i^k}$ and the upper bound $\overline{\mu_i^k}$ are constants given to the optimizer before iteration k. The advantage of this strategy comes from the special treatment of side constraints available in *Conlin* that makes them computationally less expensive than general linear constraint (see [41]). However, in order to prevent the optimizer from violating Eq. (4.5) and to avoid flipping of the variables, the side constraints expression has to be more conservative than the original linear constraints. We choose to compute the bounds as follows:

$$\underline{\mu_i^k} = \begin{cases} \mu_i^{k-1} - \frac{\mu_i^{k-1} - \mu_{i+1}^{k-1}}{2} & \text{if} \quad 1 \leqslant i \leqslant n_r - 1, \\ \mu_{min} & \text{if} \quad i = n_r, \end{cases}$$

$$\overline{\mu_i^k} = \begin{cases} 1 & \text{if} \quad i = 1, \\ \mu_i^{k-1} + \frac{\mu_{i-1}^{k-1} - \mu_i^{k-1}}{2} & \text{if} \quad 2 \leqslant i \leqslant n_r. \end{cases}$$

Then, the drawback of this zero order approximation of Eq. (4.5) is a strong reduction of the admissible design space. As a result, the variables updates by the optimizer become smaller, and the convergence of the complete optimization procedure becomes slower. Also, the optimization process has a strong tendency to get locked into intermediate density designs. To avoid this situation, a continuation procedure that gradually increases penalty up to very high values (up to $p = 48$) has to be used in order to reach 0-1 material distribution.

Special purpose optimizer for topology (*Spot*)

Alternatively, newer and 'confidential' developments of *Conlin* software include a special treatment of the constraints of the kind $\mu_j \leqslant \mu_i$. In this new module called *Spot*, these constraints can be handled directly at the dual problem level without requiring addition of classical linear constraints.

As explained in previous sections, problem reformulation and manufacturing constraint approximation failed to solve the issue in an efficient way. Therefore, the use of *Spot* and its dedicated treatment of this manufacturing constraint seems to be necessary even so we have to use it as a black box. Nevertheless, to validate the approach implemented in *Spot*, its results have been compared with the ones obtained when solving the complete optimization problem using linear constraints. The final structure provided by both methods is almost identical and the objective function final values differ only by $10^{-3}\%$. But computational time is by far lower with *Spot*. Over 200 iterations, the average time needed by *Spot* is of 0.03 s while on average the classic approach requires 9.5 s with

peaks up to 200 s. Moreover, conversely to the side constraint approximation, with *Spot* a penalty value $p = 5$ is sufficient to get a 0-1 material distribution. Therefore, the numerical applications proposed in next sections have been solved with *Spot*.

4.4.2 Numerical applications

Clamped-clamped microbeam suspension optimization

To illustrate the influence of the manufacturing constraints, we have chosen to study the two examples already treated in Section 4.3. At first, we are going to consider the introduction of the manufacturing constraints into the clamped-clamped microbeam suspension design problem (see Figure 4.4 for dimensions and boundary conditions). However, when the new constraint is activated, convergence to a 0-1 material distribution turned out to be difficult. As we can see in Figure 4.12 with penalty parameter $p = 3$, large areas of intermediate densities are remaining. In our opinion, this issue arises from the high sensitivities existing in the elements of the top of the domain (which results in the arch in Figure 4.5). Therefore, the optimizer prefers to place intermediate density material higher in the domain rather than placing unitary densities on the imposed layer of material.

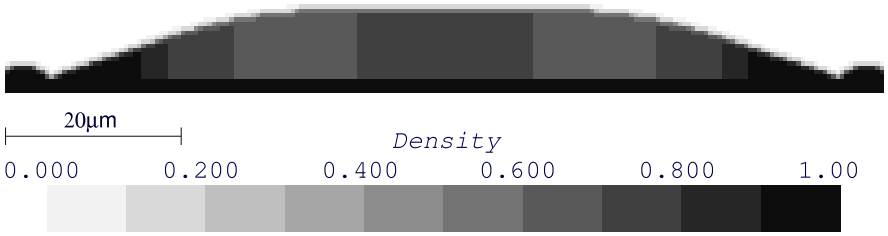


Figure 4.12: Convergence issue when manufacturing constraint is enabled.

To reach 0-1 material distribution, we had to set $p = 5$ from the first iteration of the optimization process. Continuation procedure on p has also been attempted but it did not produce satisfactory results. Indeed, at the beginning of the optimization while p is low, the optimizer tends to get locked into a design similar to the one presented in Figure 4.12. From there, it is more difficult to converge to a 0-1 distribution as it requires p to be higher than 5.

Moreover, we noticed that a uniform initial material distribution can slow down convergence of the optimization process. Indeed, because of the manufacturing constraint, with a uniform material distribution only the lowest and the highest row of design variables are free while all intermediate rows are locked. Consequently, the optimization process converges very slowly and more than 400 iterations are needed to reach convergence for the clamped-clamped microbeam application. Therefore, we choose to replace it by a new initial material distribution where densities are gradually decreasing in the upward direction. This way, all variables are initially free. The initial distribution used for present application is presented in Figure 4.13.

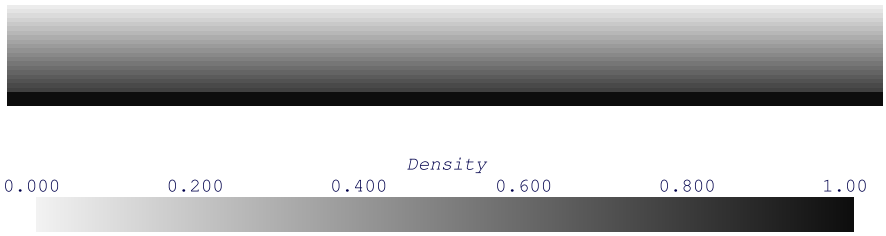
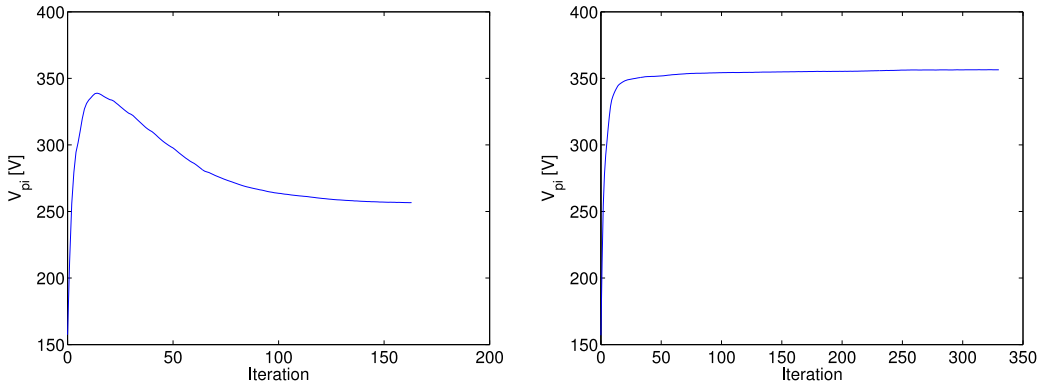


Figure 4.13: Gradient initial distribution.

Finally, while the sensitivity filter does not cause any problem when manufacturing constraint is not present, it turns out that it is incompatible with the latter. Indeed, when combining sensitivity filter with the manufacturing constraint, pull-in voltage starts to decrease monotonously after a few iterations as shown in Figure 4.14(a). This



(a) Sensitivity filtering.

(b) Density filtering.

Figure 4.14: Pull-in voltage history for clamped-clamped microbeam optimization with manufacturing constraint.

behavior is completely unexpected since the objective is to maximize pull-in voltage. After investigation, we have noticed that disabling the heuristic sensitivity filter solved the problem. Nevertheless, as mesh dependency problems can occur if the topology optimization problem is not regularized, we choose to replace the sensitivity filter by a density filter (see Eq. (3.3)). Conversely to sensitivity filter, the density filtering technique is rigorous and does not rely on heuristic modifications of the sensitivities. Consequently, the sensitivities are now consistent with the objective function behavior and we get the objective function history plotted in Figure 4.14(b).

Starting with the gradient initial distribution and penalty $p = 5$, we get the structure presented in Figure 4.15 as solution of the clamped-clamped microbeam suspension optimization. The optimization procedure needed 330 iterations to converge and reach a final pull-in voltage equal to 356 V (see Figure 4.14(b)). The effect of the manufacturing constraint is clearly visible since no closed cavities are included in the microbeam final design. Nevertheless, the introduction of this constraint reduces by 47% the optimal

pull-in voltage.

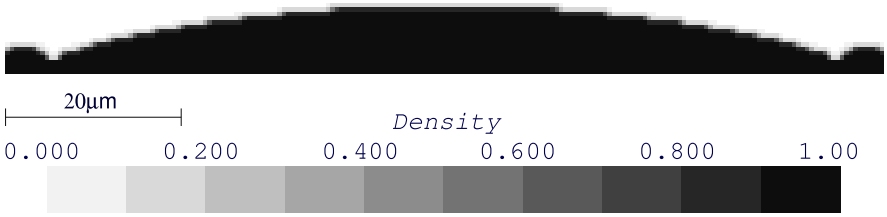


Figure 4.15: Clamped-clamped microbeam final structure with manufacturing constraint.

Moreover, this final design can be compared to the results obtained by Abdalla *et al.* [1] for thickness optimization of microbeams. The present application corresponds to the simply supported-simply supported (SS) case from Abdalla whose result is presented in Figure 3.12. Let us remind that Abdalla *et al.* use a shape optimization procedure in which only the thickness of the beam is optimized while the neutral fiber remains horizontal. Therefore Abdalla's results are symmetrical horizontally even if the variable thickness of the beam is not taken into account for electrostatic force computation. Conversely, we use a more rigorous modeling procedure and in the present optimization problem the interface between electrostatic and mechanical domain is fixed. Nevertheless, if we omit the two small bumps located at both ends of our design, the thickness distribution of the two designs are similar.

Microbeam suspension optimization with off-center electrode

The impact of the manufacturing constraint on the microbeam with off-center electrode has also been investigated. The boundary conditions and the dimensions are identical to those in Figure 4.7. However, the optimization problem is slightly different, the available material volume is doubled and is now equal to 40% of the design domain volume. Moreover, the penalty parameter is fixed to 5 from the first iteration instead of the continuation procedure.

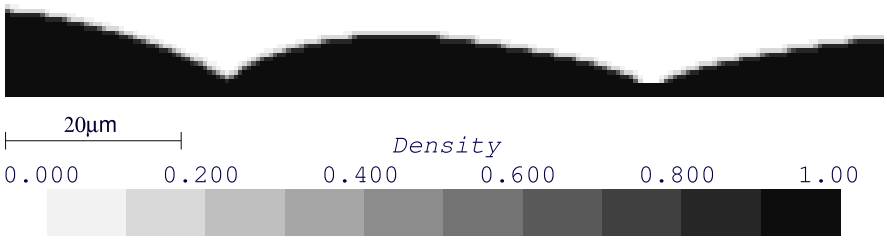


Figure 4.16: Microbeam with off-center electrode final structure with manufacturing constraint.

The final structure is presented in Figure 4.16. Obviously, the structure presents the same asymmetry as the one obtained without the manufacturing constraint; design

material is gathered on the left side because electrostatic forces only act on the left half of the microbeam. Besides, in spite of the twice larger amount of design material the final pull-in voltage is still lower than without the constraint. It is equal here to 723 V i.e. 20% less.

With respect to the work proposed by Abdalla *et al.* , even if the electrostatic pressure is different in this case, the final structure is again similar in terms of thickness distribution (see Figure 3.12, clamped-clamped boundary conditions). Moreover, both methods lead to a structure that presents two hinges (i.e. minimum thickness points) and look like designs obtained in other optimization problems involving stability as described by Seyranian and Privalova [97] for post-buckling optimization of columns.

4.5 Conclusion

The simplified optimization problem considered along this chapter is a first step towards the development of an optimization procedure enabling to control stability of microelectromechanical microdevices. The simplification of the optimization problem consists in defining a partition of the modeling domain that insulates the optimization domain from electrostatic effects. By consequence, electrostatic forces application points are fixed and the optimization domain is purely mechanical.

Under this assumption, path following algorithm, topology optimization and monolithic element formulation have been combined to establish a maximization procedure of the pull-in voltage. The optimization procedure is also based on semi-analytical sensitivities of pull-in voltage. The capabilities of the developed method are illustrated with two numerical examples and the accuracy of the semi-analytic sensitivities is verified using finite differences. Moreover, even if the assumptions associated with the simplified problem reduce strongly the freedom of the optimizer, comparison of pull-in maximization results with compliance minimization shows that the two optimization criteria are not strictly equivalent.

Moreover, in the scope of this simplified optimization problem, it has been possible to include a manufacturing constraint to broaden the manufacturing possibilities of the optimum designs. The manufacturing constraint prevents the optimizer from creating closed holes in the structure. Therefore the final designs are more compliant with the classical MEMS manufacturing techniques based on thin layer deposition and etching.

The numerical applications proposed in this chapter demonstrate the possibility to control pull-in voltage using topology optimization in the framework of a simplified optimization problem. However, the hypotheses underlying the simplification of the optimization problem restrict strongly the design freedom. Indeed, in sight of optimizing electrostatic actuators, the optimizer has to be able to move the electrostatic force application points. Therefore, the suppression of the simplifying assumptions and the generalization of the optimization problem is the topic of a later chapter. Never-

theless, before generalizing the optimization problem, next chapter will investigate the possibility of an alternative formulation for pull-in voltage based optimization.

Chapter 5

Eigenproblem formulation for the simplified optimization problem

5.1 Introduction

When pull-in effect has been described in Chapter 2, it has been noted that this behavior is very similar to nonlinear buckling. Consequently, the aim of the present chapter is to exploit this similarity to enhance pull-in optimization with existing techniques in buckling. In this way, an alternative approach to pull-in optimization is proposed.

Topology optimization of structures subject to buckling has been studied thoroughly in the past. At first, linear buckling has been considered. The first application of linear buckling optimization was proposed by Neves *et al.* [73]. Then several articles have contributed to this topic, see for instance Refs. [26, 59, 65]. The general framework to optimize buckling behavior relies on the following stability eigenproblem that allows computing the buckling load:

$$(\mathbf{K}_e + \lambda \mathbf{K}_\sigma)\,\mathbf{p} = \mathbf{0}, \tag{5.1}$$

where $\mathbf{K}_e$ is the linear stiffness matrix and $\mathbf{K}_\sigma$ the geometric stiffness matrix (also called stress matrix) as defined in Ref. [49, Part 2, Chap. 7]. The geometric stiffness matrix is computed on the basis of the structural linear response to a reference load $\mathbf{f}$. The value of the buckling load is then equal to the reference load multiplied by the lowest eigenvalue of (5.1). To improve the stability of the structure, Neves *et al.* [73] formulate the optimization problem as the maximization of the lowest eigenvalue of Eq. (5.1).

However, earlier work on buckling analysis by Brendel and Ramm [23] shows that linear prediction of the buckling load provided by Eq. (5.1) usually overestimates the buckling load. Therefore, a lot of efforts have been made to include non-linear behavior into the optimization problem. Wu and Arora [121] derive the nonlinear buckling load sensitivities in case this limit load corresponds to a limit point. Next, two formulations based on a *one-point* and a *two-point* linearization of the eigenvalue problem have been

developed and are summarized by Park and Choi [80]. The two methods are based on the principle that the determinant of the tangent stiffness matrix vanishes at limit and bifurcation points.

The *one-point* formulation initially developed by Brendel and Ramm [23] consists in dividing the tangent stiffness matrix $\mathbf{K}_t$ according to the different contributions:

$$\mathbf{K}_t = \mathbf{K}_e + \mathbf{K}_u + \mathbf{K}_\sigma, \tag{5.2}$$

with the initial displacement matrix $\mathbf{K}_u$. From Eq. (5.2), Brendel and Ramm define two eigenproblems which allow evaluating the instability load:

$$(\mathbf{K}_e + \mathbf{K}_u + \lambda \mathbf{K}_\sigma)\, \mathbf{p} = \mathbf{0}, \tag{5.3}$$

$$(\mathbf{K}_e + \lambda\, (\mathbf{K}_u + \mathbf{K}_\sigma))\, \mathbf{p} = \mathbf{0}. \tag{5.4}$$

The first eigenproblem in Eq. (5.3) considers that the initial displacement matrix is independent of the applied load. If this eigenproblem is formulated at a very low load level where the initial displacements are very small, it is equivalent to the linear buckling eigenproblem from Eq. (5.1). Eq. (5.4) presents another eigenproblem formulation where the initial displacements are supposed to scale with the load. The results provided by the two versions of these *one-point* eigenproblems are studied by Brendel and Ramm [23]. The two eigenproblems are solved on several prebuckling points of the nonlinear equilibrium path. It turns out that the results provided by both formulations are more accurate than using the linear eigenproblem. Also, the predicted buckling load becomes more and more accurate as we are approaching the actual buckling load.

Rather than trying to guess which part of the tangent stiffness matrix depends on the load level, Bathe and Dvorkin [10] propose to use the value of the tangent stiffness matrix at two successive load levels to formulate the following *two-point* eigenproblem:

$$\left(\mathbf{K}_t^{i-1} + \Delta\lambda\, \left(\mathbf{K}_t^{i} - \mathbf{K}_t^{i-1}\right)\right) \mathbf{p} = \mathbf{0}, \tag{5.5}$$

where $\mathbf{K}_t^i$ denotes the tangent stiffness matrix at path following point i as show in Figure 5.1. If the path following points i and $i-1$ are close to each other, the *two-point* formulation provides automatically a good estimate of the variation of $\mathbf{K}_t$ with respect to the external load.

Notice that with the eigenproblem (5.5), a different notation is used for the eigenvalue ($\Delta\lambda$) since it has a different meaning than in the *one-point* formulation and the linear problem. Indeed, in Eq. (5.5) $\Delta\lambda$ corresponds to an increment of the load and not to the load itself as in Eqs. (5.1), (5.3) and (5.4). At buckling point, the eigenvalue is equal to 1 for both formulations (i.e. if i correspond to the buckling state for the *two-point* formulation).

Sensitivity analysis of the estimated critical loads provided by *one-point* and *two-point* formulation is developed by Park and Choi [80]. However, the sensitivities obtained with their method do not converge toward the sensitivities of the actual buckling load. This difficulty is circumvented by Kwon *et al.* [55] by using the exact sensitivity expressions.

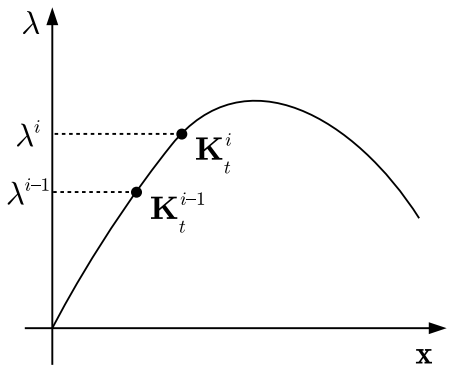


Figure 5.1: Linear extrapolation of the tangent stiffness matrix.

Recently an optimization procedure for composite structures based on a *one-point* approach has been proposed by Lindgaard *et al.* [59–62]. The method is dedicated to wind turbine blade fiber angle optimization and aims to maximize the buckling load. Mode crossing difficulties are avoided by including in the optimization problem not only the first eigenvalue from Eq. (5.3) but also higher order eigenvalues. This last idea allows including into the optimization process higher (i.e. upcoming) instability modes which are not reachable by following the equilibrium curve.

5.2 Eigenproblem formulation

The developments described above for nonlinear buckling present a lot of interest in the electromechanical pull-in context. Indeed, as a nonlinear stability problem, pull-in effect is very similar to nonlinear buckling and identical issues like mode switching appear as illustrated in the numerical examples of this chapter. The ability to compute and to optimize estimated pull-in voltages is very advantageous and provides an alternative to the rigorous pull-in voltage optimization presented in Section 4.2.

However, the definition of a linearized eigenproblem for electromechanical microdevices is not straightforward. As we have seen, several definitions of the linearized eigenproblem have been defined for nonlinear buckling. This section is dedicated to the choice of a suitable eigenproblem. At first, a *one-point* and a modified *two-point* formulations are attempted on basis of the original tangent stiffness matrix. Secondly, a *two-point* eigenproblem is established using a condensed version of the tangent stiffness matrix.

5.2.1 One-point formulation

The objective of the present section is to define a stability eigenproblem following an approach similar to the one proposed by Lindgaard *et al.* [59]. This means that we are going to try to separate blocks of the tangent stiffness matrix $\mathbf{K}_t$ that depend on the load parameter ($\mathbf{K}_t^{\lambda}$) from the one that do not ($\mathbf{K}_t^{0}$) so that we can write the linear

approximation:

$$\mathbf{K}_t \approx \mathbf{K}_t^0 + \lambda \mathbf{K}_t^\lambda,$$

and define the following eigenproblem to look for singular points of $\mathbf{K}_t$ and the associated load multipliers λ_k:

$$\left(\mathbf{K}_t^0 + \lambda_k \mathbf{K}_t^\lambda\right) \mathbf{p}_k = \mathbf{0}.$$

The current procedure corresponds to a one-point formulation [23] as it resorts to information from one equilibrium point. Let us first recall the expression of the electromechanical tangent stiffness matrix as a block matrix:

$$\begin{aligned} \mathbf{K}_t &= \begin{bmatrix} \mathbf{K}_{\phi\phi}(\mathbf{u}) & \mathbf{K}_{\phi u}(\mathbf{u}, \boldsymbol{\phi}) \\ \mathbf{K}_{u\phi}(\mathbf{u}, \boldsymbol{\phi}) & \mathbf{K}_{uu}(\mathbf{u}, \boldsymbol{\phi}) \end{bmatrix}, \\ &= \begin{bmatrix} \mathbf{K}_{\phi\phi}(\mathbf{u}) & \mathbf{K}_{\phi u}(\mathbf{u}, \boldsymbol{\phi}) \\ \mathbf{K}_{u\phi}(\mathbf{u}, \boldsymbol{\phi}) & \mathbf{K}_{uu}^0 + \mathbf{K}_{uu}^*(\mathbf{u}, \boldsymbol{\phi}) \end{bmatrix}. \end{aligned} \tag{5.6}$$

The following blocks appear:

- $\mathbf{K}_{\phi\phi}$ is the 'stiffness' matrix of the electrostatic problem, it solely depends on mechanical displacements ($\mathbf{u}$).
- $\mathbf{K}_{\phi u}$ and $\mathbf{K}_{u\phi}$ are the coupling matrices. They depend on both displacement and voltage distribution.
- $\mathbf{K}_{uu}^0$ is the mechanical stiffness matrix; it is constant in the linear case.
- $\mathbf{K}_{uu}^*$ is the electrostatic force contribution to mechanical stiffness, it depends on both displacement and voltage distribution.

According to the dependencies of each block, the tangent stiffness matrix can be split in two ways. The first one is similar to the way Lindgaard established his eigenproblem by keeping in $\mathbf{K}_t^0$ all blocks that do not depend explicitly on the load factor. By proceeding this way, we get the following eigenvalue problem:

$$\left(\begin{bmatrix} \mathbf{K}_{\phi\phi}(\mathbf{u}) & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{uu}^0 \end{bmatrix} + \lambda_k \begin{bmatrix} \mathbf{0} & \mathbf{K}_{\phi u}(\mathbf{u}, \boldsymbol{\phi}) \\ \mathbf{K}_{u\phi}(\mathbf{u}, \boldsymbol{\phi}) & \mathbf{K}_{uu}^*(\mathbf{u}, \boldsymbol{\phi}) \end{bmatrix} \right) \mathbf{p}_k = \mathbf{0}. \tag{5.7}$$

However, one may object that displacements are also dependent on applied voltage and that therefore, $\mathbf{K}_{\phi\phi}$ should also be included into $\mathbf{K}_t^\lambda$. However, this block should not be simply moved from one matrix to the other one since $\mathbf{K}_{\phi\phi}(\mathbf{u} = 0)$ is different from zero. It is then better to include the variation of $\mathbf{K}_{\phi\phi}$ with respect to the initial state into matrix $\mathbf{K}_t^\lambda$. This would result in the following eigenproblem:

$$\left(\begin{bmatrix} \mathbf{K}_{\phi\phi}(\mathbf{0}) & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_{uu}^0 \end{bmatrix} + \lambda_k \begin{bmatrix} \mathbf{K}_{\phi\phi}(\mathbf{u}) - \mathbf{K}_{\phi\phi}(\mathbf{0}) & \mathbf{K}_{\phi u}(\mathbf{u}, \boldsymbol{\phi}) \\ \mathbf{K}_{u\phi}(\mathbf{u}, \boldsymbol{\phi}) & \mathbf{K}_{uu}^*(\mathbf{u}, \boldsymbol{\phi}) \end{bmatrix} \right) \mathbf{p}_k = \mathbf{0}. \tag{5.8}$$

Moreover, by keeping $\mathbf{K}_{\phi\phi}(\mathbf{0})$ in $\mathbf{K}_t^0$ we avoid singularity of this matrix.

In the two eigenproblems formulated above (Eq. (5.7) and (5.8)), the second term matrix $\mathbf{K}_t^{\lambda}$ corresponds to the load applied to the system and also to the part of the tangent stiffness matrix $\mathbf{K}_t$ which is expected to vary with a modification of the applied voltage. The load variable λ acts as a multiplier of the load. By solving the eigenproblem, we look for multipliers values λ_k that lead to instability (i.e. singularity of $\mathbf{K}_t$). If the eigenproblem is solved at pull-in point we should find at least one $\lambda_k = 1$. Therefore, if the eigenproblem is solved for a point on the equilibrium curve $(\mathbf{q}_i, V_i)$, we suggest computing the predicted pull-in voltages using:

$$V_{pi,pred,k} = \lambda_k V_i.$$

In principle, we can expect that all λ_k are greater than or equal to one so that $V_{pi,pred,k} \geqslant V_i$. Otherwise it means that an instability point has been passed or that the instability voltage has been exceeded which may happen in case of a bifurcation. Moreover, the smallest value of $V_{pi,pred,k}$ provides an estimate of the actual pull-in voltage of the structure.

Application to a 1D problem

To evaluate the capabilities of the two eigenproblem formulations described above, they have been applied to the 1D system presented in Figure 5.2. However, as it comes up that both eigenproblems lead to similar conclusions, only the results provided by the second formulation (Eq. (5.8)) are presented here below.

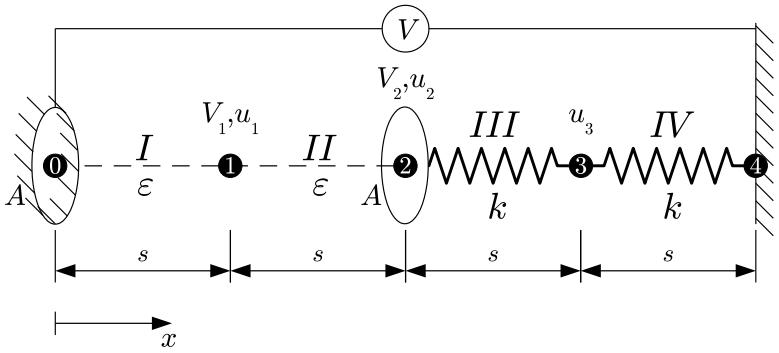


Figure 5.2: Sketch of the 1D system.

The studied system in Figure 5.2 is an idealization of an infinite plane capacitor which possesses a fixed and an elastically suspended electrode. The rigid electrode is on the left at node 0, the voltage is imposed to zero on this electrode. The elastic electrode is suspended by elements III and IV, each element having a stiffness k and being considered as a perfect conductor. Therefore, the voltage at node 2, V_2 is equal to the voltage over the whole mobile electrode: $V_2 = V$ the imposed voltage. The electrostatic domain is composed of elements I and II, it is filled with a nonstructural dielectric material of permittivity ε. Despite the elements I and II are purely electrostatic, a displacement

variable u_1 is affected to node 1. Actually, variable u_1 does not correspond to a degree of freedom. It allows adapting the electrostatic domain mesh to the displacement of the mechanical part to avoid twist (i.e. degeneracy) of element II. There are two mechanical degrees of freedom: u_2 and u_3 respectively the displacement of nodes 2 and 3. While the only electrostatic degree of freedom is the voltage at node 1: V_1.

The modal problem in Eqs. (5.7) and (5.8) are based on the tangent stiffness matrix of the system. The finite element approach described in Ref. [90, Chap. 5] has been applied to establish the following tangent stiffness matrix:

$$\mathbf{K}_t = \begin{bmatrix} \overbrace{-\left(\frac{\varepsilon A}{x_1}+\frac{\varepsilon A}{x_2-x_1}\right)}^{\mathbf{K}_{\phi\phi}} & \overbrace{-\varepsilon A\frac{V_2-V_1}{(x_2-x_1)^2} \quad 0}^{\mathbf{K}_{\phi u}} \\ \underbrace{-\varepsilon A\frac{V_2-V_1}{(x_2-x_1)^2}}_{\mathbf{K}_{u\phi}} & \underbrace{k-\left(\frac{V_2-V_1}{x_2-x_1}\right)^2\frac{\varepsilon A}{x_2-x_1} \quad -k}_{\mathbf{K}_{uu}} \\ 0 & -k \quad 2k \end{bmatrix} \begin{matrix} V_1 \\ u_2 \\ u_3 \end{matrix} . \tag{5.9}$$

In this equation, parameters x_1 and x_2 denote the absolute position of nodes 1 and 2 respectively. They are related to u_1 and u_2 degrees of freedom by the following relations:

$$x_1 = s + u_1, \qquad x_2 = 2s + u_2.$$

The different sub-matrices included in the tangent stiffness matrix are highlighted in Eq. (5.9) according to their physical meaning (see Eq. (5.6)). On the right side of the matrix, the corresponding degree of freedom is indicated for each line of the matrix.

Since eigenproblems (5.7) and (5.8) are formulated on the equilibrium curve, we can express the value of V_1 as a function of V_2, x_1 and x_2. Indeed as the two electrostatic elements are made of the same material, the voltage simply decreases linearly from node 2 to node 0 and we get:

$$V_1 = x_1\frac{V_2}{x_2},$$

and by inserting this relation in the tangent stiffness matrix, one gets:

$$\mathbf{K}_t = \begin{bmatrix} -\left(\frac{\varepsilon A}{x_1}+\frac{\varepsilon A}{x_2-x_1}\right) & -\varepsilon A\frac{V_2}{x_2(x_2-x_1)} & 0 \\ -\varepsilon A\frac{V_2}{x_2(x_2-x_1)} & k-\frac{\varepsilon A}{x_2-x_1}\left(\frac{V_2}{x_2}\right)^2 & k \\ 0 & k & 2k \end{bmatrix} . \tag{5.10}$$

In this matrix, V_2 is imposed (it is the input voltage V) and x_1 depends on the mesh adaptation strategy. Therefore the only unknown parameter is x_2. The tangent stiffness matrix of the 1D system being given by Eq. (5.10) we are able to write explicitly the

eigenproblem from Eq. (5.8):

$$\left(\begin{bmatrix} -\frac{2\varepsilon}{s} & 0 & 0 \\ 0 & k & k \\ 0 & k & 2k \end{bmatrix} + \lambda \begin{bmatrix} -\left(\frac{\varepsilon A}{x_1} + \frac{\varepsilon A}{x_2 - x_1} - \frac{2\varepsilon A}{s}\right) & -\varepsilon A \frac{V_2}{x_2(x_2-x_1)} & 0 \\ -\varepsilon A \frac{V_2}{x_2(x_2-x_1)} & -\frac{\varepsilon A}{x_2-x_1}\left(\frac{V_2}{x_2}\right)^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right) \mathbf{p} = \mathbf{0}. \tag{5.11}$$

As one can see, the second term matrix $\mathbf{K}_t^{\lambda}$ is singular. This is not an issue on numerical point of view for most eigenvalue solvers but this means that one of the three eigenvalues of the problem is always infinite.

With the present 1D system, there is only one possible pull-in mode. Consequently, it is pointless to try to detect other upcoming instabilities. Therefore the eigenproblem (5.11) has been assessed on its ability to predict the instability voltage from points located on the stable part of the equilibrium curve (i.e. between rest position and pull-in position). In practice, a sequence of 100 points is defined from $x_2 = 2s$ to $x_2 = 1.2s$ and the eigenproblem (5.11) is solved on each point. This sequence includes the pull-in point which arises for $x_2 = 1.33s$.

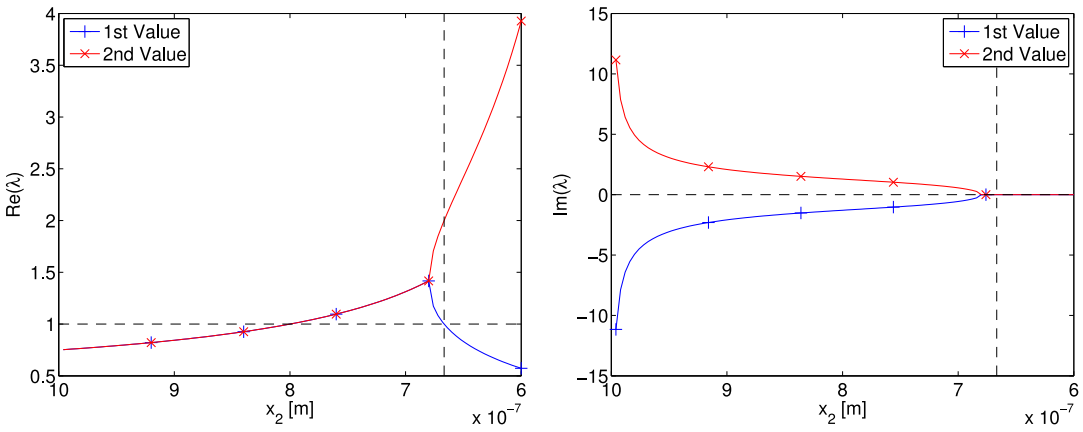
	Value
s	$0.5 \cdot 10^{-6}$ m
A	10^{-10} m^2
ε	$8.85 \cdot 10^{-12}$ F/m
k	80 N/m

Table 5.1: 1D system parameters values.

The numerical values used for the 1D system parameters are summarized in Table 5.1. Figure 5.3 plots the two finite eigenvalues of problem (5.11) for varying x_2. A major difference arises here with respect to the result obtained by Lindgaard *et al.* [59] with nonlinear buckling. Indeed, as shown by the plot in Figure 5.3(b), our problem definition leads to complex eigenvalues.

In the two plots, the rest position is located on the left side of the x_2 axis at $x_2 = 10^{-6}$, the pull-in position is represented by the vertical dashed line in Figure 5.3(a). By comparing the two plots, we can see that the two eigenvalues are complex conjugate from rest position. Then, when displacement is increased, their complex part decreases up to the point where the two eigenvalues merge and become real. Just after merging the two eigenvalues bifurcate with one increasing and the other decreasing but then staying real. As expected, the first eigenvalue is equal to 1 at pull-in point.

Similar observations have been done when using the alternative formulation (5.7). The existence of complex eigenvalues is difficult to interpret on a physical point of view. Indeed, multiplying the load matrix by a complex value has no physical meaning in the present framework. However, mathematically it is possible to explain why complex eigenvalues arise. In fact, it is possible to show (see Ref. [81, Chap. 15]) that for the



(a) Real part.

(b) Imaginary part.

Figure 5.3: Variation of the two first eigenvalues along equilibrium curve.

generalized eigenproblem:

$$(\mathbf{A} - \lambda \mathbf{M})\,\mathbf{x} = \mathbf{0}.$$

We can guarantee that eigenvalues are real only if the two following conditions are satisfied:

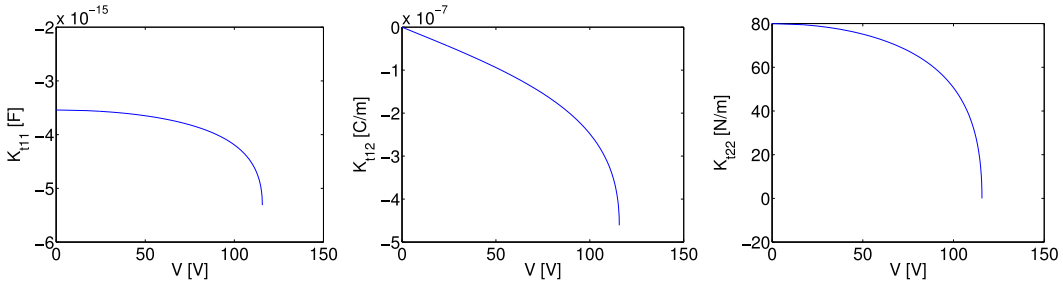
- $\mathbf{A}$ and $\mathbf{M}$ are symmetric,
- $\mathbf{M}$ is positive definite.

If the first condition is verified for our problems (5.7) and (5.8), it is not true for the second one since $\mathbf{K}_t^{\lambda}$ is indefinite in both cases. This comes from the fact that the multiphysic tangent stiffness matrix is also indefinite. Moreover some blocks of the matrix may change definiteness along the equilibrium curve like $\mathbf{K}_{uu}$ which is positive definite at rest position and becomes indefinite when approaching pull-in. This makes difficult the formulation of an eigenproblem by splitting the tangent stiffness matrix in two parts. Therefore we have to conclude that unfortunately the present approach does not provide easily exploitable results.

5.2.2 Two-point formulation

The last example, and more particularly, the obtained tangent stiffness matrix (see Eq. (5.10)) show that some elements of the tangent stiffness matrix are strongly dependent on the position of the system on the equilibrium curve. The variation of the non-constant elements along the equilibrium curve is plotted in Figure 5.4.

These plots suggest that the method used in the last section may not be appropriate in our case even though it has been successfully used by Lindgaard *et al.* [59]. Indeed, the *one-point* formulation consists in sorting out the different contributions to the tangent



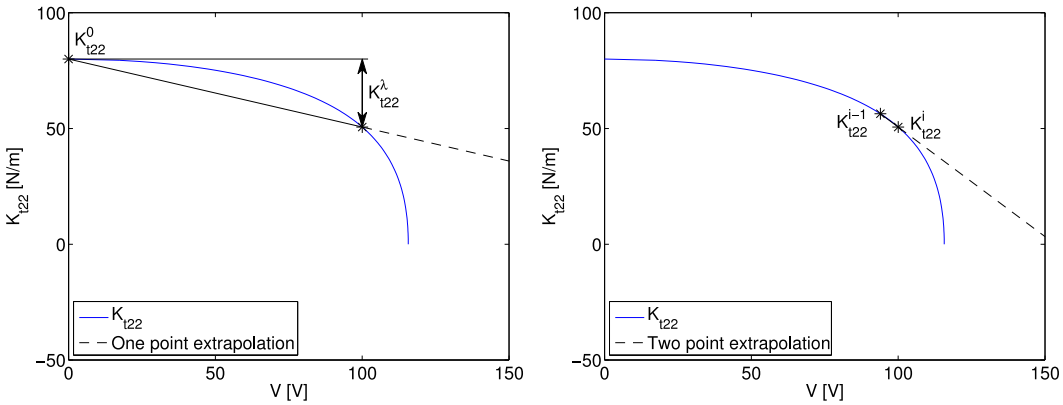
(a) $K_{t11} = -\left(\frac{\varepsilon A}{x_1} + \frac{\varepsilon A}{x_2 - x_1}\right)$. (b) $K_{t12} = -\frac{\varepsilon A V_2}{x_2(x_2 - x_1)}$. (c) $K_{t22} = k - \frac{\varepsilon A}{x_2 - x_1}\left(\frac{V_2}{x_2}\right)^2$.

Figure 5.4: Variation of $\mathbf{K}_t$ matrix elements (see Eq. (5.10)) with respect to applied voltage.

stiffness matrix into the (supposed) constant part $\mathbf{K}_t^0$ and the variable part which simply scales with the load variable $\mathbf{K}_t^\lambda$ to formulate the following approximation:

$$\mathbf{K}_t \approx \mathbf{K}_t^0 + \lambda \mathbf{K}_t^\lambda.$$

If we consider the case of $\mathbf{K}_{uu}$, $\mathbf{K}_t^0$ corresponds to the initial stiffness at rest (the mechanical stiffness) while $\mathbf{K}_t^\lambda$ contains the electrostatic force contribution to the stiffness (see Eqs. (5.7) and (5.8)). An example of the resulting extrapolation is presented in Figure 5.5(a) by the dashed line. In this example, $\mathbf{K}_t^\lambda$ is evaluated for V=100 V. As one can see, because of the nonlinearity of $\mathbf{K}_{uu}$, the extrapolation is very poor. Considering the two other tangent stiffness matrix elements plotted in Figure 5.4 similar results are expected for these elements.



(a) *One-point* extrapolation. (b) *Two-point* extrapolation.

Figure 5.5: Illustration of *one-point* and *two-point* estimates.

Conversely, the *two-point* formulation proposed by Bathe *et al.* [10] makes use of two consecutive equilibrium points to build up a local approximation and is more suited to approximate the matrix variation as it is visible in Figure 5.5(b). The definition of the eigenproblem we are going to use is slightly different from the original formulation pro-

posed by Bathe. It consists in linearizing $\mathbf{K}_t$ at current equilibrium point by computing its derivative with respect to the load variable λ using finite differences:

$$\mathbf{K}_t\left(\lambda^i + \Delta\lambda\right) \approx \mathbf{K}_t^i + \Delta\lambda \underbrace{\frac{\mathbf{K}_t^i - \mathbf{K}_t^{i-1}}{\lambda^i - \lambda^{i-1}}}_{\mathbf{K}_t'^i}, \tag{5.12}$$

where $\mathbf{K}_t^i = \mathbf{K}_t\left(\lambda^i\right)$ and the i superscript denotes successive continuation points on the equilibrium curve as shown in Figure 5.1. The main difference with Bathe's definition is the division of $\mathbf{K}_t^i - \mathbf{K}_t^{i-1}$ by $\lambda^i - \lambda^{i-1}$. Moreover, we are using the point i as starting point and we use backward finite differences to compute the derivative while Bathe *et al.* use $i-1$ and uses forward difference to compute the variation of $\mathbf{K}_t$.

Our objective is to predict instability loads, which means the λ values such that:

$$\det\left(\mathbf{K}_t\left(\lambda\right)\right) = 0.$$

An estimate of these load values can be found by using the linearization proposed in Eq. (5.12) to write the following eigenproblem:

$$\left(\mathbf{K}_t^i + \Delta\lambda_k \mathbf{K}_t'^i\right)\mathbf{p}_k = \mathbf{0}, \tag{5.13}$$

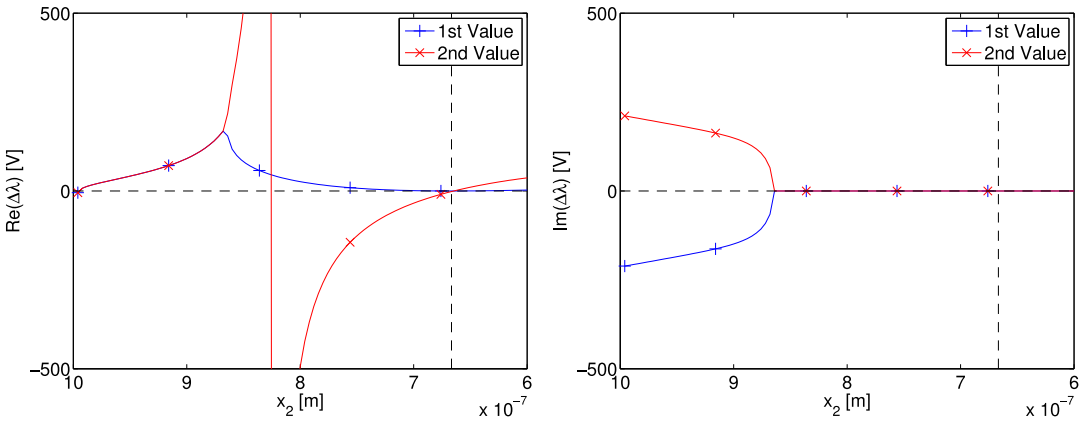
whose eigenvalues $\Delta\lambda_k$ are the load variable increments leading to instability. As we use backward finite differences the first eigenvalue is equal to zero when the eigenproblem is solved at pull-in point.

Conversely to the eigenproblem formulation from previous section (Eqs. (5.7) and (5.8)), the eigenproblem provides here critical *increments* of the load variable instead of critical *multiplier* of the load. Previously with Eqs. (5.7) and (5.8), provided a real value of λ, the instability voltage would be estimated by λV^i. Now, in eigenproblem Eq. (5.13), λ acts as an auxiliary load variable. Therefore, it is needed to link the load auxiliary variable λ to the physical load variable V (i.e. to define λ as a function of V).

Application to a 1D problem

For the present formulation let us simply define $\lambda(V) = V$ and test as previously the eigenproblem (5.13) on the 1D problem described in Figure 5.2. A more precise study of the relation between λ and V is performed in the next section but is not needed here. Figure 5.6 plots the two first eigenvalues from Eq. (5.13) against the equilibrium displacement. Once again, one can notice that the two eigenvalues are complex conjugate at the beginning of the path following. Numerical verification shows that, the matrix $\mathbf{K}_t'^i$ in eigenproblem (5.13) is indefinite over this part of the equilibrium path.

In comparison with the *one-point* formulation in Eq. (5.8), the two eigenvalues provided by the present *two-point* eigenproblem become real earlier (i.e. for lower value of the



(a) Real part. (b) Imaginary part.

Figure 5.6: Variation of the two first eigenvalues along equilibrium curve.

displacement). Moreover, both eigenvalues equal zero at pull-in conversely to the *one-point* formulation. However, right before pull-in an eigenvalue is negative which makes the present approach difficult to use in a practical case. Indeed, as we set $\lambda = V$, the predicted instability voltages are computed as:

$$V_{pi,pred,k} = V^i + \Delta\lambda_k.$$

Consequently, negative eigenvalues suggest that there is an instability voltage lower than the current voltage which is of course impossible in the present simple example. With an actual optimization which involves a large number of degrees of freedom, such results may be difficult to interpret.

5.2.3 Condensation of the tangent stiffness matrix

The monolithic formulation leads to an indefinite tangent stiffness matrix in the sense that it is neither positive nor negative semi-definite. This is related to the fact that the global tangent stiffness matrix mixes a positive definite matrix $\mathbf{K}_{uu}$ (at least for a portion of the equilibrium curve) and a negative definite matrix $\mathbf{K}_{\phi\phi}$. As it is illustrated in the last section, this indefinite characteristic makes a modal analysis more complicated as the matrix has both negative and positive eigenvalues. This situation does not appear in nonlinear buckling (see Ref. [59]) and makes the present problem more complicated. The method developed by Lindgaard *et al.* would be more easily applied to our problem if we could establish a semi-definite matrix on basis of the global tangent stiffness matrix while still preserving its physical relevance.

This is possible by applying a condensation of the electrostatic degrees of freedom on the mechanical part of the matrix. If we use the tangent stiffness block subdivision from Eq. (5.6), the condensation is written:

$$\mathbf{K}_C = \mathbf{K}_{uu} - \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}. \tag{5.14}$$

The condensation of the electrostatic part on the mechanical degrees of freedom ensures that the $\mathbf{K}_C$ keeps a physical meaning. For instance, the matrix $\mathbf{K}_C$ is also singular at pull-in point while it is not the case for $\mathbf{K}_{uu}$. Moreover, $\mathbf{K}_C$ has the advantage to be positive definite before pull-in point. In fact this matrix is very similar to a nonlinear buckling stiffness matrix.

Using $\mathbf{K}_C$, the most suited way to define an eigenvalue problem is to use a *two-point* linearization on the basis of finite differences as in Eq. (5.12). Indeed splitting the matrix expression (5.14) looks difficult because firstly both terms depend on the applied voltage V and secondly the dependence of the second term in V is probably highly nonlinear as it involves the inverse of $\mathbf{K}_{\phi\phi}$. Therefore, we suggest approximating $\mathbf{K}_C$ as follows:

$$\mathbf{K}_C\left(\lambda^i + \Delta\lambda\right) \approx \mathbf{K}_C^i + \Delta\lambda \frac{d\mathbf{K}_C^i}{d\lambda} \approx \mathbf{K}_C^i + \Delta\lambda \frac{\mathbf{K}_C^i - \mathbf{K}_C^{i-1}}{\lambda^i - \lambda^{i-1}}. \tag{5.15}$$

The last equation is expressed from a point $(\lambda_i, \mathbf{q})$ of the equilibrium curve in order to extrapolate the behavior of the tangent stiffness matrix. In the remaining of this chapter, we call the evaluation voltage (noted V_{eig}) the voltage corresponding to the load level λ_i at which the linear approximation is formulated. Starting from this first order approximation, we can define the following eigenproblem:

$$\left(\mathbf{K}_C^i + \delta \frac{d\mathbf{K}_C^i}{d\lambda}\right)\mathbf{p} = \mathbf{0}, \tag{5.16}$$

where we have replaced $\Delta\lambda$ by δ to get a classical eigenproblem formulation. As the matrix $\mathbf{K}_C$ is singular at pull-in point, the eigenproblem allows computing the load factor increment that would lead to instability (pull-in). Eigenproblem (5.16) is formulated at a given load level λ_i, the corresponding voltage is called the evaluation voltage and noted V_{eig}. The next step in the development of the method is to define the expression of λ as a function of the applied voltage; this is the scope of the next subsection.

Load variable choice

In the case of buckling modal analysis the load variable is linearly linked to the forces applied on the structure as it is presented by Lindgaard [59]. The load variable acts as a multiplier of the loads applied to the structure. In the present case, the input voltage V is the analog of the external forces in buckling. Therefore it would make sense to use also a linear relation between V and the load variable λ by simply choosing $\lambda(V) = V$ (and $\Delta\lambda = \Delta V$).

However, as it has already been mentioned, Eq. (5.16) can be interpreted as a linear approximation of $\mathbf{K}_C$. The quality of this linear approximation and of the predicted instability load is of course influenced by the choice of the relation between λ and V. Or stated differently, there may be choices of $\lambda(V)$ which make $\mathbf{K}_C(\lambda)$ more linear than others. Consequently, the relation between λ and V has been chosen on the basis of the study of a 1D system presented in Figure 5.2.

With the tangent stiffness matrix established in Eq. (5.10), we can evaluate $\mathbf{K}_C$ by using its definition in Eq. (5.14). First, the inverse of $\mathbf{K}_{\phi\phi}$ is simply:

$$\mathbf{K}_{\phi\phi}^{-1} = \left[\frac{-x_1\left(x_2 - x_1\right)}{\varepsilon A x_2}\right].$$

Which allows computing the second term of the condensation equation:

$$\mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u} = \begin{bmatrix} \frac{-\varepsilon A V_2^2 x_1}{\varepsilon x_2^3 (x_2 - x_1)} & 0 \\ 0 & 0 \end{bmatrix}.$$

As one can see, this matrix is sparse. In a realistic model it is also the case since nonzero elements only appears for nodes which belong to both mechanical and electrical finite elements. Finally, we can write the expression of the condensed matrix:

$$\mathbf{K}_C = \begin{bmatrix} k - \frac{\varepsilon A V_2^2}{x_2^3} & -k \\ -k & 2k \end{bmatrix}. \tag{5.17}$$

As said above, an obvious choice for the definition of the load variable λ is simply to set $\lambda\left(V\right) = V$. This direct equality is analog to the load variable definition used in buckling. However, if we look at $\mathbf{K}_C$ expression in Eq. (5.17), we can see that V_2 (which is equal to the input voltage V) appears squared in the first diagonal term. This means that $\mathbf{K}_C$ rather evolves proportionally to V^2 alike electrostatic force. Therefore setting $\lambda\left(V\right) = V^2$ is another interesting possibility.

Before comparing the two selected load variable choices, let us remark that x_2 is also present in $\mathbf{K}_C$ expression. As this variable represents the elastic electrode extremity position, x_2 also depends on the applied voltage V. Consequently, it would be interesting to substitute x_2 by its expression as a function of V in order to get a more accurate idea of the dependence of $\mathbf{K}_C$ in V. Nevertheless, the forces equilibrium leads to the following equilibrium equation:

$$V = \sqrt{\frac{k x_2^2\left(2s - x_2\right)}{\varepsilon A}}.$$

As one can see, inverting the last equation is not trivial and even if possible would not lead to a simple explicit expression of x_2. Therefore, we choose to neglect the dependence of K_C in x_2 for the choice of the load variable. This assumption is similar to the one done by Lindgaard [59] where the global displacement stiffness matrix is supposed to be independent of loading for the formulation of the eigenproblem.

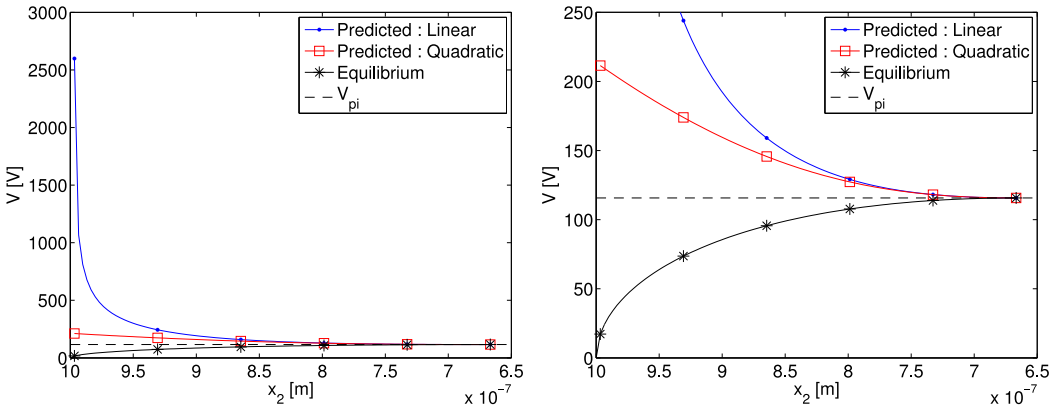
In order to choose between $\lambda\left(V\right) = V$ and $\lambda\left(V\right) = V^2$, we propose to compare the actual instability voltage level with the estimated pull-in voltages provided by the two

alternative definitions of the load variable. We have therefore two eigenproblem formulations:

Linear: $\lambda(V) = V$	Quadratic: $\lambda(V) = V^2$
$\left(\mathbf{K}_C^i + \delta_{lin}\frac{d\mathbf{K}_C^i}{dV}\right)\mathbf{p} = \mathbf{0}$	$\left(\mathbf{K}_C^i + \delta_{quad}\frac{d\mathbf{K}_C^i}{dV^2}\right)\mathbf{p} = \mathbf{0}$
$V_{pi,lin} = V^i + \delta_{lin}$	$V_{pi,quad} = \sqrt{(V^i)^2 + \delta_{quad}}$

(5.18)

where V^i is the voltage at which the eigenproblem is formulated. In both cases, if the eigenproblem is formulated at pull-in point, the eigenvalues δ_* are equal to zero and the estimated pull-in voltage is exact. However, as we want to predict instability points that are not reachable by following the equilibrium curve, we are rather interested in seeing how the two eigenproblems behave when solved prior to the pull-in point. That's why the two formulations are compared along the equilibrium curve in Figure 5.7 using the 1D system from Figure 5.2 with the properties given in Table 5.1. With these parameters, the device pull-in voltage is 115.70 V. In Figure 5.7, the right plot is a zoom of the left one. The two predicted voltage curves are obtained by following the equilibrium curve from rest position up to pull-in configuration. To be as close as possible from an actual optimization problem conditions, finite differences have been used to evaluate the derivatives of $\mathbf{K}_C$ involved in the two eigenproblems formulations (5.18) even if in the present example an analytic expression is available. The actual pull-in voltage is materialized by the horizontal dashed line and the star labeled line represents the equilibrium curve.



(a) Linear and quadratic prediction.

(b) Zoom on quadratic prediction.

Figure 5.7: Predicted instability voltage for linear and quadratic derivation.

The two predicted pull-in voltage curves start slightly after rest position at $x_2 = 9.97 \cdot 10^{-7}$ m (rest position is at $x_2 = 10^{-6}$ m) where setting $\lambda(V) = V$ leads to very strong overestimation. Then, when decreasing x_2 (following the horizontal axis to the right) the two predictions get closer to the actual pull-in voltage to finally coincide at

pull-in point ($x_2 = 6.67 \cdot 10^{-7}$ m).

All along the equilibrium curve, the two formulations overestimate pull-in voltage. Nevertheless, the quadratic definition of the load variable provides significantly better results than the linear definition. This difference is particularly noticeable far from the instability point at the beginning of the equilibrium curve. Indeed on the first computed point the relative difference between the predicted instability voltage and the actual one is of 83% for the quadratic definition and of 2146% for the linear one. Moreover, because of the expression of $\mathbf{K}_C$ matrix and of the dependence of electrostatic force in V^2 the quadratic definition is physically more relevant than the linear one. It comes that this definition $\lambda = V^2$ will be used in the following.

Analogy with aeroelasticity

There is another physically based eigenproblem where the eigenvalues are related to the square of the physical quantity representing the load. The divergence phenomenon known in aeroelasticity (see Ref. [34] for a complete description) is another stability problem that can be solved by formulating an eigenproblem.

Briefly, the divergence effect can be explained as follows, when traveling into a fluid, a wing is submitted to a lift force and also to a longitudinal torsion moment. These loads result in deformations of the wing and among them a longitudinal torsion of the wing appears. This longitudinal torsion causes a modification of the angle of attack along the wing which in turn generally results in an increase of the aerodynamics lift and torsion moment. With those increasing loads, the wing deformation increases again. As a consequence, there is a limit speed called divergence speed above which the system becomes unstable and the deformation is unbounded.

Basic theory developed in Ref. [34] leads to an eigenproblem that allows determining the divergence speed. Because the aerodynamic forces are proportional to the square of the airspeed, it comes out that the eigenvalues are also proportional to the square of the airspeed.

5.3 Double actuator model study

Since the way to define the eigenproblem is now determined, we are going to test it on a simple benchmark in order to evaluate the ability to detect upcoming instabilities. The benchmark is represented in Figure 5.8. This model represents two electrostatic actuators submitted to the same input voltage. From a mechanical point of view, the two actuators are completely uncoupled. Their dimensions and physical properties are identical except the stiffness of the elastic electrode. Actuator 1 stiffness is equal to k while actuator 2 stiffness is $k + \Delta k > k$. The higher stiffness for actuator 2 leads to a higher pull-in voltage for this actuator. The two actuators do not interact with each other, neither mechanically or electrically, they only share the same input voltage.

The pull-in voltage of each actuator can be computed if the two devices are modeled separately. However, if the two actuators are considered together, it is not possible with our analysis procedure to compute pull-in state for actuator 2. Indeed actuator 1 reaches pull-in before actuator 2 and at this point the only possibility to preserve equilibrium is to decrease the applied voltage. As a result, when the two actuators are modeled together with our modeling tools, the pull-in point of actuator 2 is unreachable.

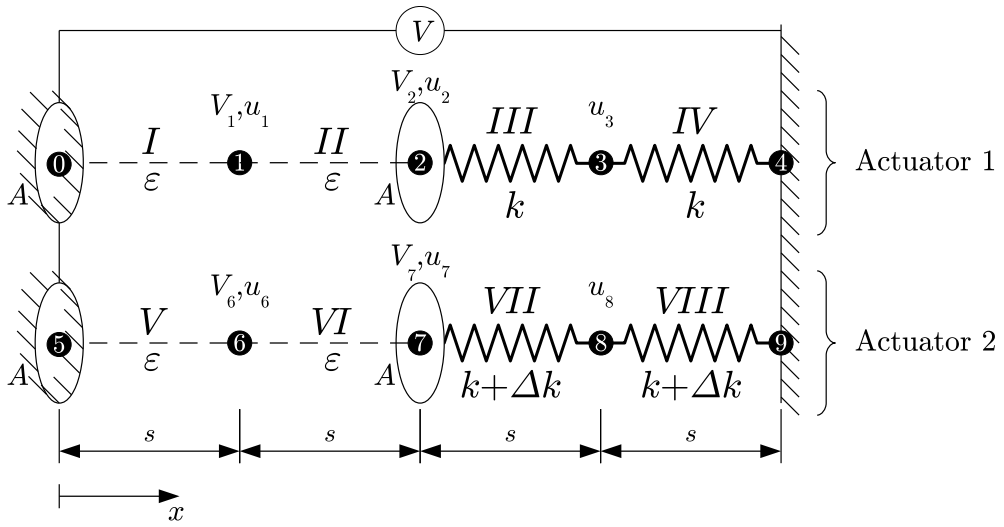


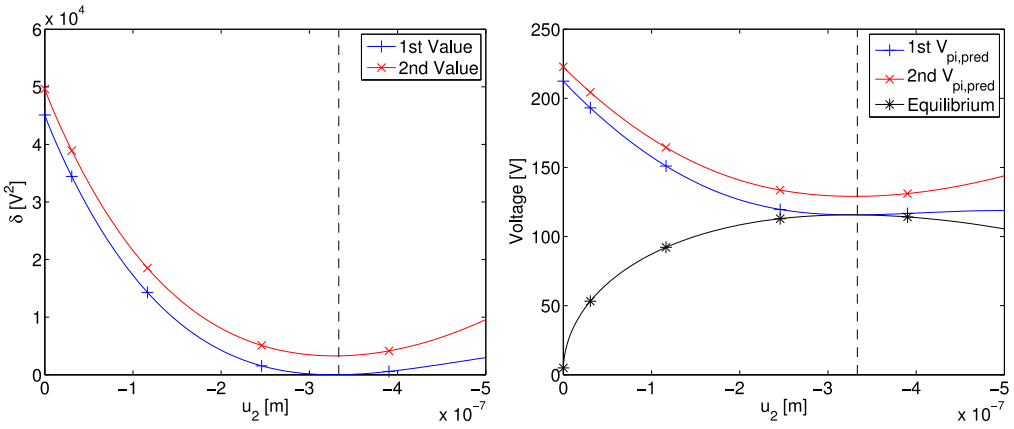
Figure 5.8: Double actuator test model.

The numerical values of the benchmark parameters are listed in Table 5.2. The pull-in voltage of each actuator taken separately is also indicated in the last line of the table. Actuator 2 is 10 % stiffer than the first one which results in a 4.9% higher pull-in voltage. The other parameters are identical. The principle of the present benchmark is similar to the one from Section 5.2.3. We are going to follow the equilibrium curve and solve the eigenproblem (5.16) along the curve. Nevertheless, in the present case a numerical continuation (normal flow algorithm) has been used rather than a simple sampling of the analytic equilibrium curve.

	Actuator 1	Actuator 2
s [m]	$0.5 \cdot 10^{-6}$	$0.5 \cdot 10^{-6}$
A [m^2]	10^{-10}	10^{-10}
ε [F/m]	$8.8542 \cdot 10^{-12}$	$8.8542 \cdot 10^{-12}$
Stiffness [N/m]	$k = 80$	$k + \Delta k = 88$
V_{pi} [V]	115.70	121.34

Table 5.2: Double actuator parameters values.

Results provided by eigenproblem (5.16) are represented in Figure 5.9. On the left, Figure 5.9(a) represents the variation of the first two eigenvalues against displacement of the first actuator tip and, on the right, Figure 5.9(b) draws the corresponding pre-



(a) Eigenvalues. (b) Predicted instability voltages.

Figure 5.9: Results provided by condensed eigenproblem along the equilibrium curve.

dicted pull-in voltages together with the equilibrium curve. As in last example from Section 5.2.3, both predicted voltage and eigenvalues decrease from rest position up to pull-in point which is represented by the vertical dashed line. This results from the eigenproblem increasing accuracy as we are approaching pull-in point. At pull-in point the lowest eigenvalue vanishes as expected while the second one remains greater than zero. If we have a look at the eigenmodes associated with these eigenvalues at pull-in point in Table 5.3, we can see that they are consistent with the eigenvalues. These modes allow us linking the eigenvalues with a part of the model. The first mode only involves actuator 1, which makes sense since it is for this actuator that pull-in is expected and since the associated eigenvalue vanishes. The second mode corresponds to a displacement of actuator 2 only. This shows that the drop of the second eigenvalue is related to an upcoming unstable behavior of actuator 2. The best pull-in voltage prediction we get for actuator 2 is 129.09 V which is 6.4% overestimated compared to the actual value for this actuator (see Table 5.2).

	u_2	u_3	u_7	u_8
Mode 1	-1	-0.5	0	0
Mode 2	0	0	-1	-0.5

Table 5.3: Eigenmodes at pull-in point.

This model presents the same characteristics as a topology optimization problem where mode switching is observed. Pull-in appears in one part of the domain while another part of the domain is also very close to instability. A small modification of stiffness can inverse the situation and give rise to a totally different pull-in mode. As it is shown in the numerical application of the coming Section 5.5.2, this mode switching behavior results in oscillations and a premature stop of the optimization procedure. The present example shows that the eigenproblem formulated on basis of the condensed stiffness matrix is an efficient tool to capture upcoming instabilities. The method is

consequently very interesting from the optimization point of view.

5.4 Sensitivity analysis

5.4.1 Eigenvalue sensitivity analysis

The eigenvalues of the stability problem developed above can be used as objective function or as constraint within an optimization problem. However, to use efficient gradient based optimizer, we need to compute the derivatives of the eigenvalues with respect to the design variables.

Let us consider that all the eigenvalues are simple. Under this condition, to obtain the sensitivity of eigenvalue δ_k, we start from the eigenproblem (5.16) and apply the same procedure as the one proposed by Seyranian *et al.* [96]. The first step consists in deriving the eigenproblem with respect to a particular design variable μ_j:

$$\frac{d}{d\mu_j}\left(\left(\mathbf{K}_C + \delta_k \mathbf{K}'_C\right)\mathbf{p}_k\right) = \mathbf{0}.$$

For the sake of conciseness, we have posed here $\mathbf{K}'_C = \frac{d\mathbf{K}_C}{d\lambda}$ and temporarily omitted the path following indexes i. Next, we transform slightly this equation to write it in canonical form using $[-\mathbf{K}'_C]$ instead of $\mathbf{K}'_C$ in order to have a positive semi-definite matrix in the second term. It comes:

$$\begin{gathered}\frac{d}{d\mu_j}\left(\left(\mathbf{K}_C - \delta_k\left[-\mathbf{K}'_C\right]\right)\mathbf{p}_k\right) = \mathbf{0},\\ \Leftrightarrow \left(\frac{d\mathbf{K}_C}{d\mu_j} - \frac{d\delta_k}{d\mu_j}\left[-\mathbf{K}'_C\right] - \delta_k\frac{d\left[-\mathbf{K}'_C\right]}{d\mu_j}\right)\mathbf{p}_k + \left(\mathbf{K}_C - \delta_k\left[-\mathbf{K}'_C\right]\right)\frac{d\mathbf{p}_k}{d\mu_j} = \mathbf{0}.\end{gathered}$$

Then we premultiply the last equation by $\mathbf{p}_k^T$:

$$\begin{gathered}\mathbf{p}_k^T\left(\frac{d\mathbf{K}_C}{d\mu_j} - \frac{d\delta_k}{d\mu_j}\left[-\mathbf{K}'_C\right] - \delta_k\frac{d\left[-\mathbf{K}'_C\right]}{d\mu_j}\right)\mathbf{p}_k + \underbrace{\mathbf{p}_k^T\left(\mathbf{K}_C - \delta_k\left[-\mathbf{K}'_C\right]\right)}_{=\mathbf{0}}\frac{d\mathbf{p}_k}{d\mu_j} = \mathbf{0},\\ \Leftrightarrow \frac{d\delta_k}{d\mu_j} = \frac{1}{\mathbf{p}_k^T\left[-\mathbf{K}'_C\right]\mathbf{p}_k}\mathbf{p}_k^T\left(\frac{d\mathbf{K}_C}{d\mu_j} - \delta_k\frac{d\left[-\mathbf{K}'_C\right]}{d\mu_j}\right)\mathbf{p}_k. \qquad (5.19)\end{gathered}$$

The usual convention is to normalize the eigenmodes $\mathbf{p}_k$ so that $\mathbf{p}_k^T\left[-\mathbf{K}'_C\right]\mathbf{p}_k = 1$ which simplifies the sensitivities expression.

The derivative of $\mathbf{K}_C$ with respect to the load variable λ is computed using finite differences in Eq. (5.15). Therefore, $\mathbf{K}'_C$ can be replaced in the sensitivity equation (5.19). This actually results in switching the derivative with respect to the design variable μ_j with the derivative with respect to the load variable and gives:

$$\frac{d\delta_k^i}{d\mu_j} = \mathbf{p}_k^{iT}\left(\frac{d\mathbf{K}_C^i}{d\mu_j} + \delta_k^i\frac{\frac{d\mathbf{K}_C^i}{d\mu_j} - \frac{d\mathbf{K}_C^{i-1}}{d\mu_j}}{\lambda_k^i - \lambda_k^{i-1}}\right)\mathbf{p}_k^i. \qquad (5.20)$$

Finally, we can regroup the derivatives of the condensed stiffness matrix:

$$\frac{d\delta_k^i}{d\mu_j} = \mathbf{p}_k^{iT}\left(\left(1+\frac{\delta_k^i}{\lambda_k^i-\lambda_k^{i-1}}\right)\frac{d\mathbf{K}_C^i}{d\mu_j} - \frac{\delta_k^i}{\lambda_k^i-\lambda_k^{i-1}}\frac{d\mathbf{K}_C^{i-1}}{d\mu_j}\right)\mathbf{p}_k^i. \tag{5.21}$$

5.4.2 Condensed stiffness matrix sensitivity

The last equation shows that evaluating the sensitivities requires providing the sensitivity of the condensed stiffness matrix at two different points of the equilibrium curve. These sensitivities can be obtained by deriving the definition of the condensed stiffness matrix (see Eq. (5.14)).

$$\begin{aligned}\frac{d\mathbf{K}_C}{d\mu_j} &= \frac{d}{d\mu_j}\left(\mathbf{K}_{uu} - \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}\right), \\ &= \frac{d\mathbf{K}_{uu}}{d\mu_j} - \frac{d\mathbf{K}_{u\phi}}{d\mu_j}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u} - \mathbf{K}_{u\phi}\frac{d\mathbf{K}_{\phi\phi}^{-1}}{d\mu_j}\mathbf{K}_{\phi u} - \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\frac{d\mathbf{K}_{\phi u}}{d\mu_j}.\end{aligned} \tag{5.22}$$

It is not possible to derive analytically the inverse of $\mathbf{K}_{\phi\phi}$. However, we can substitute this term by noting that:

$$\mathbf{K}_{\phi\phi}\frac{d\mathbf{K}_{\phi\phi}^{-1}}{d\mu_j} = \underbrace{\frac{d\left(\mathbf{K}_{\phi\phi}\mathbf{K}_{\phi\phi}^{-1}\right)}{d\mu_j}}_{=\mathbf{0}} - \frac{d\mathbf{K}_{\phi\phi}}{d\mu_j}\mathbf{K}_{\phi\phi}^{-1} \quad \Leftrightarrow \quad \frac{d\mathbf{K}_{\phi\phi}^{-1}}{d\mu_j} = -\mathbf{K}_{\phi\phi}^{-1}\frac{d\mathbf{K}_{\phi\phi}}{d\mu_j}\mathbf{K}_{\phi\phi}^{-1}. \tag{5.23}$$

By inserting Eq. (5.23) into the expression (5.22) of the sensitivities of $\mathbf{K}_C$ we get, after some algebra:

$$\frac{d\mathbf{K}_C}{d\mu_j} = \frac{d\mathbf{K}_{uu}}{d\mu_j} - \frac{d\mathbf{K}_{u\phi}}{d\mu_j}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u} - \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\left(\frac{d\mathbf{K}_{\phi u}}{d\mu_j} - \frac{d\mathbf{K}_{\phi\phi}}{d\mu_j}\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}\right). \tag{5.24}$$

This expression gives us the total derivative of $\mathbf{K}_C$ as a function of the total derivatives of the tangent stiffness matrix. If we assume that the evaluation voltage V_{eig} (i.e. the load level at which the eigenproblem is formulated) is constant all over the optimization problem, a modification of the design parameters will affect the tangent stiffness matrix in two ways:

- Modification of the material properties (e.g. local modification of the stiffness).
- Modification of the generalized displacements $\mathbf{q}$ for an imposed voltage V_{eig}.

Therefore, in the present case, we can apply the following chain rule:

$$\frac{d\mathbf{K}_t}{d\mu_j} = \frac{\partial\mathbf{K}_t}{\partial\mu_j} + \frac{\partial\mathbf{K}_t}{\partial\mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}.$$

Inserting this chain rule into Eq. (5.24) leads to:

$$\begin{aligned}\frac{d\mathbf{K}_C}{d\mu_j} =& \frac{\partial \mathbf{K}_{uu}}{\partial \mu_j} + \frac{\partial \mathbf{K}_{uu}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j} \\ &- \left(\underbrace{\frac{\partial \mathbf{K}_{u\phi}}{\partial \mu_j}} + \frac{\partial \mathbf{K}_{u\phi}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}\right)\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u} \\ &- \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\left(\underbrace{\frac{\partial \mathbf{K}_{\phi u}}{\partial \mu_j}} + \frac{\partial \mathbf{K}_{\phi u}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j} - \left(\underbrace{\frac{\partial \mathbf{K}_{\phi\phi}}{\partial \mu_j}} + \frac{\partial \mathbf{K}_{\phi\phi}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}\right)\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}\right). \end{aligned} \tag{5.25}$$

In the last expression, the underlined terms vanish if the optimization domain is purely mechanic (i.e. if pseudo-density variation does not affect material electric properties). For this particular case, we get a shorter expression of sensitivity:

$$\begin{aligned}\frac{d\mathbf{K}_C}{d\mu_j} =& \frac{\partial \mathbf{K}_{uu}}{\partial \mu_j} + \frac{\partial \mathbf{K}_{uu}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j} \\ &- \left(\frac{\partial \mathbf{K}_{u\phi}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}\right)\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u} \\ &- \mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}\left(\frac{\partial \mathbf{K}_{\phi u}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j} - \left(\frac{\partial \mathbf{K}_{\phi\phi}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}\right)\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}\right). \end{aligned} \tag{5.26}$$

From a computational point of view, the factor $\mathbf{K}_{\phi\phi}^{-1}\mathbf{K}_{\phi u}$ appears several times in Eqs. (5.25) and (5.26). This term can possibly be recovered from a previous computation of $\mathbf{K}_C$ or at least can be computed only once per sensitivity analysis. Also as the tangent stiffness matrix is symmetric, $\mathbf{K}_{u\phi}\mathbf{K}_{\phi\phi}^{-1}$ does not need to be computed.

In the two equations, two different types of differential terms appear. The first type is $\frac{\partial \mathbf{K}_{xx}}{\partial \mu_j}$. These terms can be computed semi-analytically using the interpolation law derivative. Terms from the second group are all of the form $\frac{\partial \mathbf{K}_{xx}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}$. These last terms are more complicated to compute since $\frac{\partial \mathbf{K}_{xx}}{\partial \mathbf{q}}$ is a third order tensor and is not easy to compute analytically.

5.4.3 Computation of $\frac{\partial \mathbf{K}_{xx}}{\partial \mathbf{q}}\frac{d\mathbf{q}}{d\mu_j}$

For several reasons, explicit computation of $\partial \mathbf{K}_{xx}/\partial \mathbf{q}$ is not convenient and we choose instead to compute directly $\partial \mathbf{K}_{xx}/\partial \mathbf{q} \cdot d\mathbf{q}/d\mu_j$. Indeed, firstly, storing the resulting third order tensor $\partial \mathbf{K}_{xx}/\partial \mathbf{q}$ would greatly increase memory requirements of the sensitivity analysis. Secondly, at present time, there is no data structure in *Oofelie* able to handle third order tensors. And finally, the generalized displacements vector $\mathbf{q}$ counts more elements than the design variable vector $\boldsymbol{\mu}$. Indeed, $\mathbf{q}$ contains at least two values per node (three for electromechanical coupling elements) while, in a classic topology optimization formulation, the length of $\boldsymbol{\mu}$ is equal to the number of elements. Since quadrangular meshes are used, $\mathbf{q}$ is at least twice longer than $\boldsymbol{\mu}$. Consequently, explicit computation of $\partial \mathbf{K}_{xx}/\partial \mathbf{q}$ would lead to a larger number of matrix assemblies and

therefore higher computational cost than computing directly $\partial \mathbf{K}_{xx}/\partial \mathbf{q} \cdot d\mathbf{q}/d\mu_j$, unless a tailored procedure is developed to take into account sparse structure of $\partial \mathbf{K}_{xx}/\partial \mathbf{q}$.

Before establishing the matrix $\partial \mathbf{K}_{xx}/\partial \mathbf{q} \cdot d\mathbf{q}/d\mu_j$, one needs to compute the total derivative of the generalized displacements with respect to the design variables $d\mathbf{q}/d\mu_j$. This total derivative can be obtained using the equilibrium equation. If $\mathbf{r}(\mathbf{q}, V, \boldsymbol{\mu})$ denotes the difference between external and internal forces vectors (i.e. the residual vector), we have at equilibrium:

$$\mathbf{r}(\mathbf{q}, V, \boldsymbol{\mu}) = \mathbf{0}.$$

Then, the derivative of this equation with respect to the design variable μ_j is:

$$\frac{\partial \mathbf{r}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_j} + \frac{\partial \mathbf{r}}{\partial V} \frac{dV}{d\mu_j} + \frac{\partial \mathbf{r}}{\partial \mu_j} = \mathbf{0}.$$

Under the assumption that V is fixed to V_{eig}, the derivative $\frac{dV}{d\mu_j}$ vanishes. Moreover, as by definition $\mathbf{K}_t = \frac{\partial \mathbf{r}}{\partial \mathbf{q}}$ we simply get the following expression for the generalized displacements sensitivity:

$$\frac{d\mathbf{q}}{d\mu_j} = -\mathbf{K}_t^{-1} \frac{\partial \mathbf{r}}{\partial \mu_j}.$$

The derivative of $\mathbf{r}$ can be computed semi-analytically at element level using the interpolation law derivative. Usually computing $\mathbf{q}$ sensitivities is considered as an expansive operation since with n design variables, it requires n linear system solutions. However, in the present case the part of the computational time required by $d\mathbf{q}/d\mu_j$ remains small compared to the global computation time of $\partial \mathbf{K}_{xx}/\partial \mathbf{q} \cdot d\mathbf{q}/d\mu_j$.

Two approaches have been developed to evaluate $\partial \mathbf{K}_{xx}/\partial \mathbf{q} \cdot d\mathbf{q}/d\mu_j$. The first and simplest one resorts to finite differences. It requires more matrix assemblies than the second one that relies on the analytic expression of the tangent stiffness matrix to obtain semi-analytic sensitivities.

Finite differences

Using finite differences is the approach chosen by Lindgaard [59]. First, a perturbation $\Delta\mathbf{q}$ of the generalized displacements vector is defined on the basis of its sensitivities:

$$\Delta\mathbf{q} = \frac{d\mathbf{q}}{d\mu_j} \Delta\mu_j.$$

Then this perturbation is used to compute the sensitivities of $\mathbf{K}_{xx}$ due to generalized displacements contribution:

$$\frac{\partial \mathbf{K}_{xx}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_j} \approx \frac{\mathbf{K}_{xx}\left(\mathbf{q} + \frac{\Delta\mathbf{q}}{2}\right) - \mathbf{K}_{xx}\left(\mathbf{q} - \frac{\Delta\mathbf{q}}{2}\right)}{\Delta\mu_j}, \tag{5.27}$$

where $\mathbf{K}_{xx}(\cdot)$ must be understood as: '$\mathbf{K}_{xx}$ as a function of $\cdot$'.

Thanks to its simplicity, this procedure is a good reference for further developments. However, in the present case, these finite differences are computationally very expensive. Indeed for each variable, we have to assemble two times the matrix $\mathbf{K}_t$. This makes for n design variables, $2n$ assemblies of the global matrix. Moreover as in Eq. (5.21) we need the sensitivities of $\mathbf{K}_C$ at two consecutive points of the equilibrium curve we need to repeat the process two times and we finally get $4n$ assemblies. Fortunately, this process could be easily parallelized for large applications.

Semi-analytic

Alternatively to finite differences approach, it is also possible to compute analytic sensitivities of the tangent stiffness matrix with respect to generalized displacements by adapting the procedure proposed by Van Miegroet [111]. For the sake of conciseness, sensitivity computations for $\mathbf{K}_{\phi\phi}$, $\mathbf{K}_{uu}$ and $\mathbf{K}_{\phi u}$ are proposed in Appendix A. The resulting sensitivity expressions allow computing $\frac{\partial \mathbf{K}_t}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\boldsymbol{\mu}_i}$ matrix analytically for a given $d\mathbf{q}/d\boldsymbol{\mu}_i$ vector.

At first glance, the availability of analytical sensitivities cuts by a factor 2 the number of global matrices assemblies with respect to finite differences (Eq. (5.27)). However, computational time required to compute $\frac{\partial \mathbf{K}_t}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\boldsymbol{\mu}_i}$ is not divided by two since sensitivity expressions are more complex and require more matrix operations than a normal tangent stiffness assembly (see Appendix A). Nevertheless, the semi-analytic approach requires 30% less time to evaluate $\frac{\partial \mathbf{K}_t}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\boldsymbol{\mu}_i}$ than finite differences.

Another advantage of the semi-analytic approach is to avoid chaining finite differences computations. Indeed, Eq. (5.20) already resorts to finite differences to compute $\mathbf{K}'_C$. To preserve numerical stability and accuracy, it is preferable to avoid successive finite differences computations.

5.5 Numerical applications

5.5.1 Optimization of a double actuator

The stability eigenproblem allows estimating the actual as well as *hidden* instability voltages. Moreover, with the eigenvalue sensitivities provided by the last section, it is possible to include these eigenvalues in the optimization problem formulation and solve it with gradient based method. Before considering a full size topology optimization problem, its capability is first illustrated with a simple optimization problem, which consists in optimizing the suspension stiffness of a double actuator similar to the one we presented previously in Figure 5.8.

The double actuator used for the optimization problem is given in Figure 5.10. There are two major differences with respect to the one from section 5.3. Firstly, the electrodes areas are here different for each actuator. This is illustrated by the circular plates

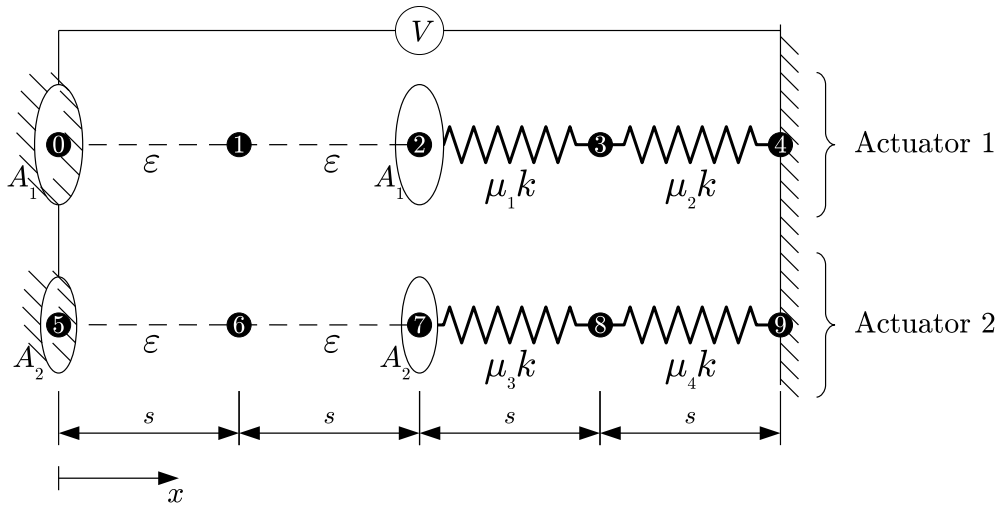


Figure 5.10: Double actuator optimization model.

of different size for each actuator. Actuator 1 electrodes are larger than the one of actuator 2. Therefore, for a given input voltage, this yields in larger electrostatic forces for actuator 1. Secondly, all the suspension springs have initially identical stiffness. The parameter values are summarized in Table 5.4. The smaller electrode area for actuator 2 leads to a larger pull-in voltage if actuator 2 would be modeled alone.

	Actuator 1	Actuator 2
s [m]	$0.5 \cdot 10^{-6}$	$0.5 \cdot 10^{-6}$
A [m^2]	10^{-10}	$0.5 \cdot 10^{-10}$
ε [F/m]	$8.8542 \cdot 10^{-12}$	$8.8542 \cdot 10^{-12}$
Stiffness [N/m]	$k = 80$	$k = 80$
V_{pi} $(\mu_i = 1)$ [V]	115.70	163.62

Table 5.4: Optimization double actuator parameters values.

The optimization problem can be seen as a sizing problem where the sections of the spring elements are adjusted in order to maximize the instability voltage. The instability voltages are estimated by using the stability eigenproblem formulation (5.16). As detailed later in this chapter, to avoid mode crossing problem (see Ref. [18]), the two lowest eigenvalues are included in the optimization problem using a max-min formulation. Initially, all design variables are equal to one. Moreover, the sum of the design variables is constrained to be less than 4 (the initial sum) as a material resource

constraint. This gives rise to the following mathematical formulation:

$$\max_{\mu} \left(\min_{k} \delta_k \right),$$
$$s.t. \begin{cases} \sum_i \mu_i \leqslant 4, \\ \mu_i \geqslant 0, \end{cases}$$
$$with \ \left(\mathbf{K}_C - \delta_k \mathbf{K}'_C \right) \mathbf{p}_k = \mathbf{0}.$$

The optimization problem is formulated on the equilibrium curve for a fixed linearization voltage of $V_{eig} = 110$ V. Ten iterations are needed for the optimizer to converge. The histories of the two estimated and of the actual pull-in voltage are plotted in Figure 5.11. As one can see, pull-in voltage is always overestimated by the stability eigenproblem method. Nevertheless, the actual pull-in voltage increases regularly during optimization, which shows that the eigenvalue sensitivities provide a satisfactory approximation of the actual pull-in voltage sensitivity.

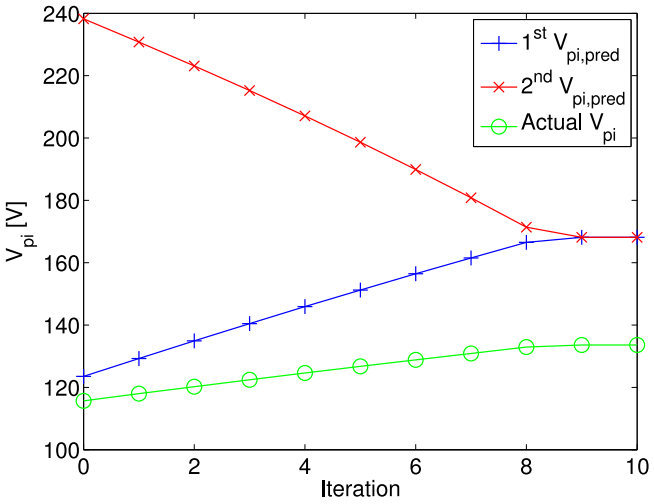


Figure 5.11: Predicted and actual pull-in voltage optimization across iterations.

The existence of a hidden instability mode is very well taken into account by the method. The two predicted pull-in voltages converge toward the same value without any oscillation. The final values of the design variables are presented in Table 5.5. At the end of the optimization process, the design variables attached to the first actuator (μ_1 and μ_2) are two times larger than the one associated with actuator 2. This counterbalances the larger area of actuator 1 electrodes and results in identical pull-in voltage value for the two actuators ($V_{pi} = 133.59$ V).

	μ_1	μ_2	μ_3	μ_4
Value	1.3333	1.3333	0.6667	0.6667

Table 5.5: Optimization variables final value.

The final design actually maximizes pull-in voltage of the double actuator. Since the material resource constraint is active, it is not possible to further increase the pull-in voltage of one of the actuator without decreasing the instability voltage of the other. Another important remark is that the final design gives rise to repeated eigenvalues as the two predicted pull-in voltages are identical (see Figure 5.11). This means that the optimization problem is not differentiable at the final point. It is not an issue in the present simple application since we know that the optimum is reached. But the possibility to obtain repeated eigenvalues has to be investigated closely for more complex problems and is considered in a future section.

5.5.2 Pull-in voltage optimization of a microbridge

This section presents numerical results provided by the pull-in voltage optimization procedure developed in Chapter 3. It illustrates why this approach may fail in some cases and why it is then mandatory to reconsider eigenproblem formulation to include several eigenvalues in the optimization problem.

Problem description

The optimization problem considered in this numerical example is presented in Figure 5.12. It consists in designing an optimal suspension for a clamped-clamped microbeam. In this example, the optimization domain (depicted in gray) is insulated from the electrostatic domain by the clamped-clamped beam (drawn in black) supposed to be perfectly conducting. Therefore electrostatic effects are not modeled in the optimization

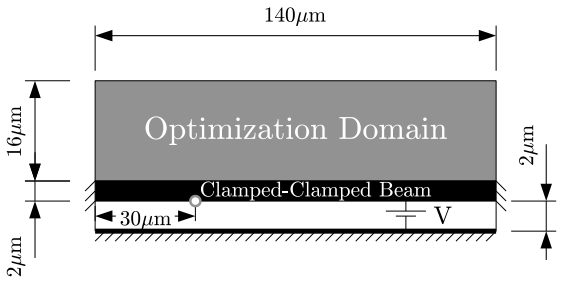


Figure 5.12: Optimization problem geometric configuration.

domain and the only material properties linked to the design variables are the elastic modulus and the density. Nevertheless, the problem is actually nonlinear because there is still a strong coupling between electrostatic and mechanical fields at the interface between the beam and the electric domain. The material used here is isotropic quartz whose physical properties are given in Table 5.6. Even though the modeling domain is symmetrical, it is completely modeled for the present application. The optimization domain is discretized using a 210 by 24 quadrangle mesh which results in 5040 design variables. The SIMP model [15] is used to represent intermediate densities. The

Young Modulus	86790 MPa
Poisson ratio	0.17
Permittivity	$39.21 \cdot 10^{-12}$ F/m

Table 5.6: Design material physical properties.

exponent parameter is set to 3 as classically chosen for SIMP.

In this section, the pull-in voltage maximization formulation is used. Therefore, the optimization problem writes:

$$\begin{aligned} &\max_{\boldsymbol{\mu}} V_{pi}\left(\boldsymbol{\mu}\right), \\ s.t. \quad &\begin{cases} v\left(\boldsymbol{\mu}\right) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i. \end{cases} \end{aligned} \tag{5.28}$$

The available material volume is set to 15% of the design domain volume. Sensitivities of the pull-in voltage V_{pi} are evaluated using pull-in configuration which is computed by following the equilibrium curve up to pull-in point using normal flow algorithm (see Chapter 2). The optimization loop is carried on until the largest modification of the design variable becomes lower than 0.01.

Optimization results starting from a uniform distribution

At first, we choose to start the optimization procedure from a uniformly gray material distribution. The design variables are all equal and satisfy exactly the volume constraint from Eq. (5.28). With this initial design, the optimization procedure converges within 92 iterations to the final structure presented in Figure 5.13. The optimized design presents a symmetrical arch suspension which increases the micro-bridge stiffness and leads to a 1066 V pull-in voltage.

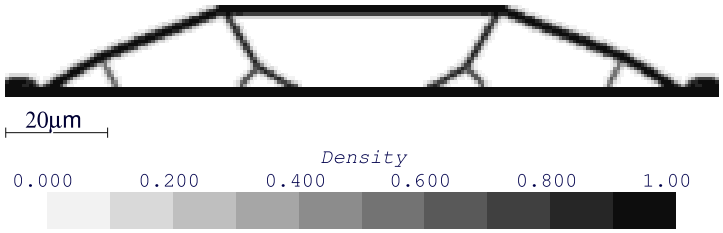
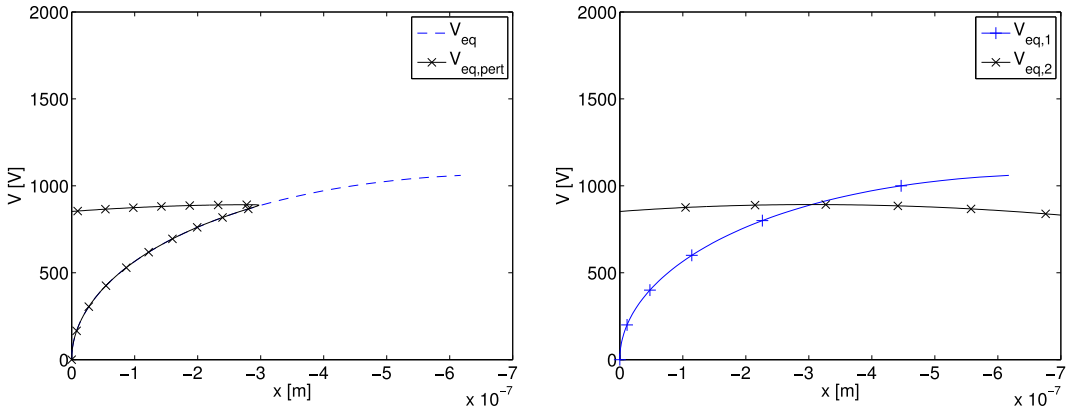


Figure 5.13: Resulting topology with pull-in voltage optimization.

Analysis

The optimized design provided by pull-in voltage optimization procedure in Figure 5.13 looks very good. However if the optimal material distribution is slightly perturbed

one may observe surprising behavior. This is illustrated in Figure 5.14(a) where two equilibrium curves are plotted in the (x, V) plane with x being the vertical displacement of the node marked by a gray circle in Figure 5.12. The dashed curve corresponds to the final design with a limit point at 1066 V. The continuous curve is the equilibrium obtained when a $5 \cdot 10^{-3}$ absolute perturbation is applied to the first design variable (located in the lower left corner of the optimization domain). When the applied voltage is lower than 880 V, the two curves are superimposed and the small design modification has no effect on the behavior of the device. However the continuous line suddenly bifurcates from the original equilibrium path at 889 V to reach a limit point for 891 V. The small design perturbation generates dramatic behavior modifications.



(a) Perturbed design equilibrium path. (b) 'Primary' and 'secondary' equilibrium curves.

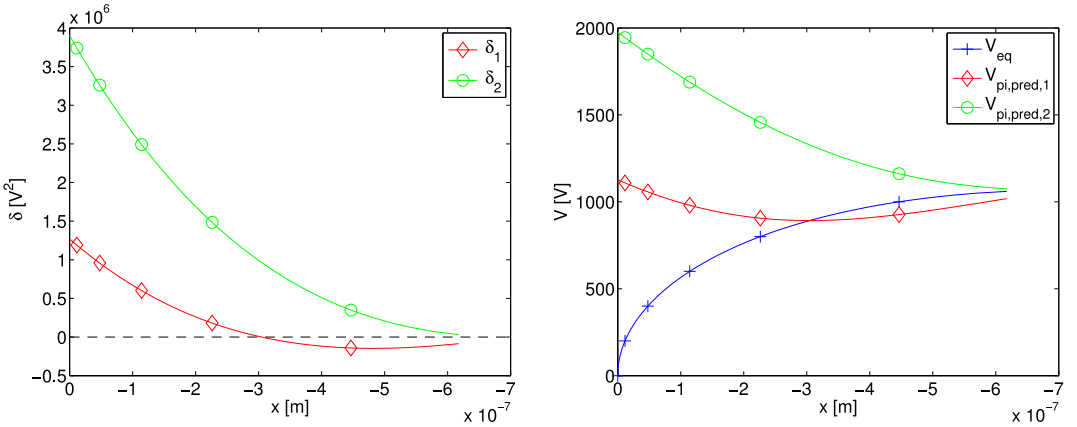
Figure 5.14: Existence of a bifurcation.

Moreover, deeper analysis shows that the bifurcation path does not only exist for the perturbed design. Indeed, as shown in Figure 5.14(b), it is possible to follow an alternative equilibrium path even for the unperturbed final design. In this figure, the blue curve is the 'primary' equilibrium path which starts from the rest position and the black curve is a 'secondary' equilibrium curve. The 'secondary' curve is obtained by initiating the continuation procedure from a post bifurcation point computed with the perturbed design. It is worth noticing an important difference between the two equilibrium paths: on the primary equilibrium path, the structure deformation is perfectly symmetrical while it is asymmetrical on the secondary (except at bifurcation point).

The stability of the 'primary' equilibrium path can be studied by computing the eigenvalues of the reduced stiffness matrix $\mathbf{K}_C$ (see Eq. (5.14)) along these curves. Along the 'primary' equilibrium curve, all eigenvalues are positive from rest position up to the bifurcation point. Then, at bifurcation point, the first eigenvalue zeroes and becomes negative after this point. The 'primary' equilibrium path is therefore stable between the rest position and the bifurcation and unstable after the bifurcation. This means that in practice even if a perfectly symmetrical structure could be manufactured, it would not be possible to follow the 'primary' equilibrium up to its limit point so that the 1066 V

pull-in voltage provided by the optimization procedure is theoretical. Considering the 'secondary' equilibrium path, it is unstable on both side of the bifurcation as the first eigenvalue is always negative.

If the bifurcation point cannot be tackled by pull-in voltage optimization procedure, the observation of $\mathbf{K}_C$ eigenvalues suggests that optimizing several eigenvalues of the stability eigenproblem (5.16) may help to cope with this issue as illustrated in Figures 5.15. Figure 5.15(a) plots the two first eigenvalues along the curve and Figure 5.15(b) represents estimated instability voltages related to the eigenvalues. These two figures show that the stability eigenproblem *is able to capture bifurcation points as well as limit points*. Indeed, the lowest instability voltage crosses the curve exactly at the bifurcation point while the second lowest instability mode tends to the limit voltage of the 'primary' path. This perspective is investigated later in this chapter.



(a) Eigenvalues against displacement. (b) Predicted pull-in voltage against displacement.

Figure 5.15: Stability analysis of the final design.

Optimization results starting from a non-uniform distribution

In literature about structural stability (see Refs. [48, 61, 118]), it is shown that bifurcation points are more likely to appear in symmetrical structures. When a bifurcation point exists, it can be transformed into a limit point by an asymmetrical modification of the design. In the last application, the choice of a symmetrical initial material distribution (uniform) leads to symmetrical deformation of the structure. As the initial deformation is symmetrical, the path following algorithm tends to keep the deformation symmetrical and misses the bifurcation point to finally reach a symmetrical limit point. Consequently, the sensitivities are symmetrical and the design variables update is also symmetrical. As the optimization progresses, it gets trapped into a symmetrical design which is not valid in practice.

To avoid the situation from last section, the same optimization problem has been solved with a different initial design point. A small random perturbation is added to the uni-

form distribution making the design asymmetrical from the beginning. The magnitude of this perturbation is 10^{-5}.

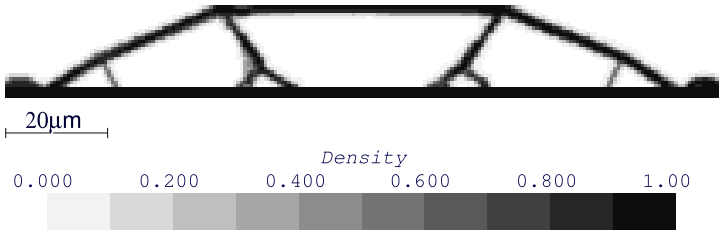


Figure 5.16: Resulting topology with pull-in voltage optimization started from a non-uniform density distribution.

The stopping criterion is satisfied after 93 iterations. The material distribution obtained at this stage is presented in Figure 5.16. Some gray areas are still visible in the figure and the layout is less clear than with a uniform initial distribution. If we consider pull-in voltage history plotted in Figure 5.17, we can see a lot of oscillations occurring after a sudden drop of the objective function. This leads us to the conclusion that the optimization process did not actually converge. The process stopped because of the adaptive move limits strategy that restricts the modification of a design variable when it starts oscillating. The final pull-in voltage value is 923 V but the highest value reached is 1045 V in iteration 31.

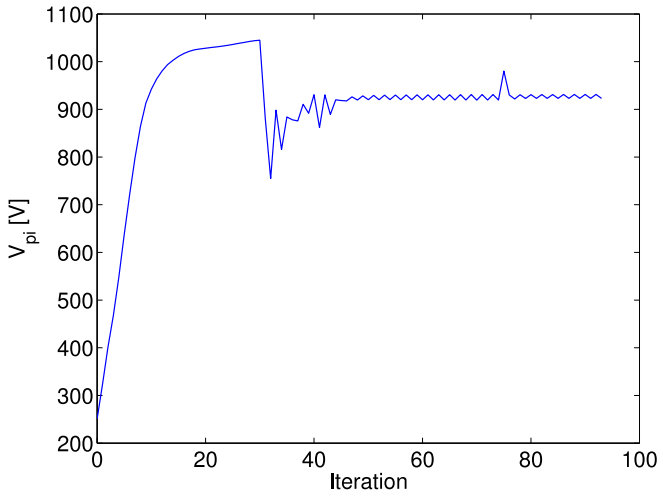
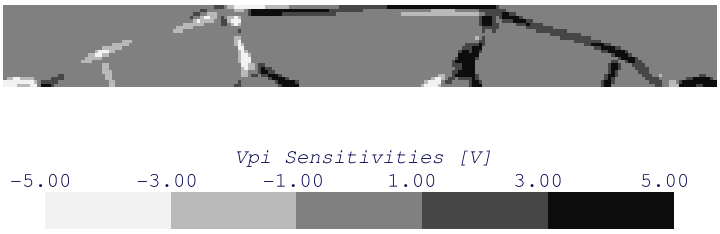


Figure 5.17: Pull-in voltage history with pull-in voltage optimization.

Analysis

To investigate the origin of these oscillations we compare the objective function sensitivity map from the two last iterations in Figure 5.18. As one can see, the two maps are very different. Indeed the maximum sensitivity area, which is the place where *Conlin*

will tend to place the material, switches from one side to the other one between two successive iterations. Such variations of the sensitivities explain the observed oscillations and suggest that the objective function is non-differentiable at this point.



(a) Iteration 92.

(b) Iteration 93.

Figure 5.18: Pull-in voltage sensitivities at two consecutive iterations.

As developed in Section 4.2.2, the expression of pull-in voltage sensitivities is the following:

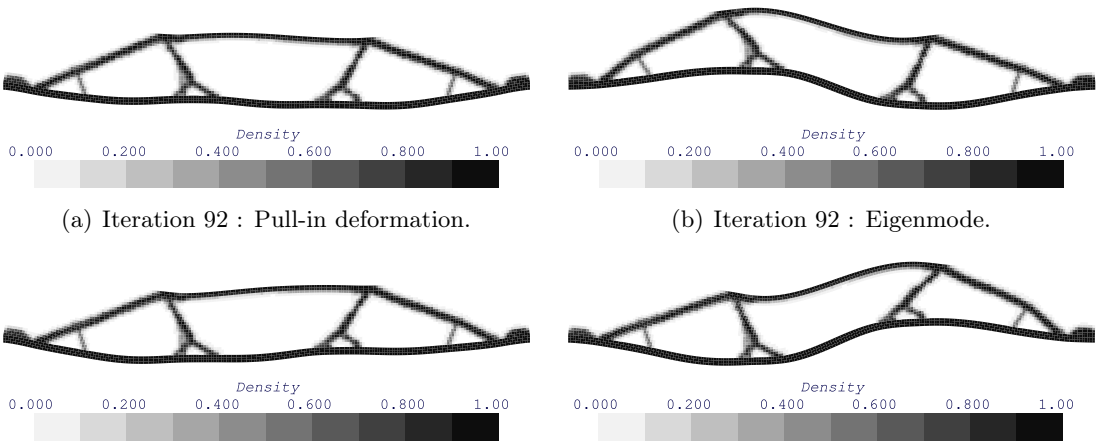
$$\frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}^T \frac{\partial \mathbf{f}_{int}}{\partial \mu_i},$$

where $\mathbf{p}$ is the first eigenmode of the tangent stiffness matrix at pull-in point and $\mathbf{f}_{int}$ stands for the internal forces. In the case of linear mechanics, this expression can be reformulated by replacing $\mathbf{f}_{int}$ by the product of the linear stiffness matrix $\mathbf{K}$ with the generalized displacements vector $\mathbf{q}$:

$$\frac{\partial V_{pi}}{\partial \mu_i} = \mathbf{p}^T \frac{\partial \mathbf{K}}{\partial \mu_i} \mathbf{q}. \tag{5.29}$$

Design variables μ_i only change slightly between the two iterations. The largest variation of the design variables is lower than 0.01 because of the move limit adaptive strategy. Therefore, the stiffness matrix derivative $\mathbf{K}$ is not prone to be at the origin of sensitivity oscillations. Conversely if we look at both eigenmodes and generalized displacements, we can see strong differences between the two consecutive iterations that lead to sudden modifications of the sensitivities.

The deformations presented in Figures 5.19(a) and 5.19(c) are magnified 10 times. We can see that in iteration 92, pull-in occurs on the right part of the structure. In iteration 93, the global deformation is similar but a closer look shows that the maximum deformation is on the left. This is confirmed by the eigenmodes $\mathbf{p}$ in Figures 5.19(b) and 5.19(d) that actually show that the pull-in location switches from the right side to the left side of the structure for a small variation of the design variables.



(a) Iteration 92 : Pull-in deformation.

(b) Iteration 92 : Eigenmode.

(c) Iteration 93 : Pull-in deformation.

(d) Iteration 93 : Eigenmode.

Figure 5.19: Deformation and mode shapes at pull-in point for two consecutive iterations.

This explains the oscillations observed during the optimization process. In iteration 92, to increase pull-in voltage, based on the sensitivities (Figure 5.18(a)) the optimizer reinforces the right side at the cost of reducing the stiffness on the left side because of the volume constraint. However, doing so, the left part becomes weaker and pull-in occurs there for the new design. Consequently, the optimizer removes material on the right side to reinforce the left side and returns to a material distribution similar to iteration 92 distribution.

This problem is similar to the 'mode-switching' issue that appears in eigenvalues optimization as described by Ma *et al.* [67]. For instance, when looking at maximization of the lowest eigenfrequency of a mechanical structure, it is not sufficient to choose the lowest eigenfrequency as objective function. Indeed, with such formulation, the initially first eigenfrequency may become greater than the second eigenfrequency after optimization. As the two first eigenfrequencies switch, their respective eigenmodes stay practically unchanged. The same phenomenon underlies the oscillations in pull-in optimization since the mode shape associated with the lowest eigenvalues changes drastically between two iterations. Fortunately, several solutions to the 'mode-switching' problem are available in literature (see for instance Refs. [53, 67]). That's why we have developed a modal approach of the pull-in maximization problem to use these formulations for pull-in voltage optimization problem.

Synthesis

This example illustrates why the pull-in optimization results should be analyzed with care. Indeed, firstly it may happen that the pull-in voltage optimization method ignores bifurcation points and lead in practice to unexpected behavior. Secondly the method may even fail to provide a converged solution. Fortunately, these two issues can be

solved by using the eigenproblem formulation from Eq. (5.16). This alternative method is able to handle both limit point and bifurcation point and several eigenvalues can be included in the optimization problem to avoid 'mode-switching'.

5.5.3 Eigenproblem formulation for pull-in voltage optimization

When dealing with eigenvalue optimization, it is more efficient to use a max-min (or alternatively min-max) formulation of the optimization problem to avoid mode-switching issue (see Refs. [53, 67]). For instance, if we consider the problem of maximizing the lowest eigenvalue δ_1 of our pull-in instability eigenproblem, we should better use the following problem:

$$\max_{\boldsymbol{\mu}} \left(\min_k \delta_k \right),$$
$$s.t. \quad g_j(\boldsymbol{\mu}) \leqslant \overline{g}_j \quad j = 1, ..., m.$$

Ideally, all eigenvalues should be included in the $\min_k \delta_k$ but for the sake of computational time, only a subset of the eigenvalues (the N_δ lowest ones) is considered. However, the actual implementation of the max-min formulation is not straightforward. Bendsøe *et al.* [18] propose to reformulate the max-min problem using a bound formulation.

$$\max_{\boldsymbol{\mu},\beta} \beta,$$
$$s.t. \begin{cases} \beta \leqslant \delta_k \quad k = 1, ..., N_\delta, \\ g_j(\boldsymbol{\mu}) \leqslant \overline{g}_j \quad j = 1, ..., m. \end{cases}$$

The bound formulation introduces an additional design parameter β. By maximizing β, the optimizer pushes all δ_k upward by adapting the physical design variables $\boldsymbol{\mu}$ to satisfy the constraints $\beta \leqslant \delta_k$. We suppose here that the δ_k are sorted in ascending order, N_δ denotes the number of eigenvalues included in the optimization problem. This bound formulation has the advantage to preserve differentiability even if mode switching occurs. An equivalent formulation is automatically used by *Conlin* whenever several objective functions are defined.

To compare the instability eigenproblem to rigorous pull-in voltage maximization, the application presented in Figure 5.12 is revisited, but this time with the following optimization problem:

$$\max_{\boldsymbol{\mu}} \left(\min_k \delta_k \right),$$
$$s.t. \begin{cases} v(\boldsymbol{\mu}) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i, \end{cases}$$

where the δ_k are the solutions of the following eigenproblem:

$$\left(\mathbf{K}_C - \delta_k \frac{d\mathbf{K}_C}{dV^2} \right) \mathbf{p}_k = \mathbf{0}. \tag{5.30}$$

This eigenproblem is solved here for a fixed voltage $V_{eig} = 245$ V all along the optimization process. V_{eig} is chosen close to the pull-in voltage of the initial structure, which is equal to 251 V so that the pull-in voltage prediction is as good as possible. The stopping criterion of the optimization is modified for the present application. Instead of using the maximum variation of the unknowns between two iterations, we preferred to fix the number of iterations to 200. Finally, the number of eigenvalues included in the optimization problem is fixed to $N_\lambda = 4$.

The topology obtained after 200 optimization iterations is presented in Figure 5.20. Despite the high number of iterations, the optimization process has not yet fully converged. This result looks different from the one provided by pull-in maximization in Figure 5.16. The result is not symmetrical and we can see that additional links appear between the imposed layer and the arch.

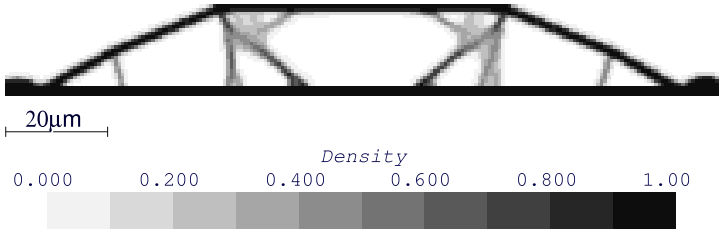
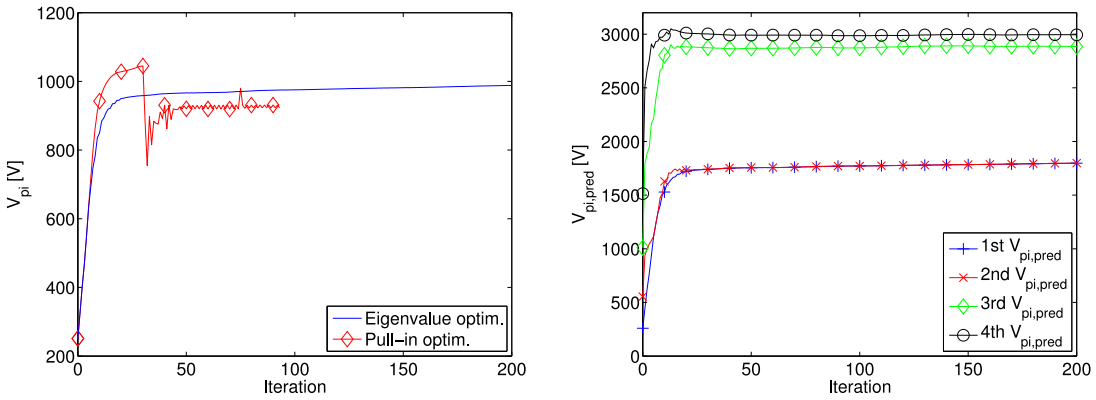


Figure 5.20: Resulting topology with stability eigenproblem optimization.

Figure 5.21(a) presents the history of the actual pull-in voltage using the eigenproblem approximation method as well as the results obtained previously with rigorous pull-in optimization. Before analyzing this curve, let us recall that pull-in voltage is not anymore the objective function of the optimization problem. We use instead a prediction or an estimate of this voltage. Therefore it is not completely rigorous to judge the efficiency of the optimization process on the basis of the actual pull-in voltage. Nevertheless, this graph allows us comparing results obtained here with those from the last section and provides us an insight into the actual performance of the optimization process.

If the material layout provided by the eigenproblem formulation is quite disappointing, pull-in voltage history is more encouraging. After 200 iterations, we get a final pull-in voltage equal to 988 V which is 7% higher than the value obtained with pull-in voltage optimization in last section when starting from non-uniform distribution. Moreover, at the very beginning of the optimization, the approximated method follows very well the increase provided by pull-in optimization method. This illustrates the consistency of the eigenproblem approximation. Pull-in voltage increases strongly during the first 25 iterations. Then it keeps on slightly increasing up to iteration 200 without actually converging. Nevertheless, the improvement of the optimization procedure comes at the cost of a computationally very expensive sensitivity analysis. The time needed for sensitivity analysis of the modal functions is of 1100 s which is very large compared to the 0.5 s required to compute sensitivities of pull-in voltage design function for the same number of design variables. The difference results from the higher complexity of



(a) Actual pull-in voltage.

(b) Predicted pull-in voltage.

Figure 5.21: Actual and predicted pull-in voltage history with stability eigenproblem optimization.

the stability eigenvalues sensitivities expression developed in Section 5.4 with respect to pull-in design function sensitivities in Eq. (5.29).

The history of the actual objective functions (the eigenvalues from problem (5.30)) is plotted in Figure 5.21(b) in terms of estimated pull-in voltage. As we have included four eigenvalues in the optimization problem, we are able to compute four predicted pull-in voltages using the following equation:

$$V_{pi,pred,k} = \sqrt{V_{eig}^2 + \delta_k}.$$

Let us, first compare the curve of the first estimated pull-in voltage $V_{pi,pred,1}$ with the actual pull-in voltage in Figure 5.21(a). On one hand, we can see that the two curves behave in a similar way across iterations with a steep increase at the beginning and a long plateau reached after 25 iterations. However, on the other hand, pull-in voltage is strongly overestimated by the eigenproblem method since at the end of the optimization, the lowest predicted pull-in voltage is equal to 1798 V (see Table 5.7) while the actual pull-in voltage is equal to 988 V. This large difference is explained by the fact that as the optimization progresses, the voltage at which the eigenproblem is formulated V_{eig} (chosen equal to 245 V here) becomes too low with respect to the actual pull-in voltage. An adaptive procedure which would modify V_{eig} during optimization would be a solution at the price of a modification of the optimization problem during optimization as presented in Section 5.7.2 at the end of this chapter.

Secondly, another interesting conclusion can be drawn by looking at the history of the four lowest predicted pull-in voltages in Figure 5.21(b). The most important one is that from the beginning of the optimization process, the first two predicted pull-in voltages are very close from each other. The two curves are even superposed on the final plateau and as Table 5.7 shows, the relative difference between the two eigenvalues is less than 0.1% so that these two eigenvalues can be considered as coalescent. As the third and

Order	Value [V^2]	Predicted V_{pi} [V]
1	3.171228e+006	1797.6
2	3.173107e+006	1798.12
3	8.261515e+006	2884.73
4	8.903429e+006	2993.92

Table 5.7: Final value of the first four eigenvalues with stability eigenproblem optimization.

the fourth eigenvalues remain higher than the two firsts, we can conclude that the first eigenvalue is double. Actually, with a double eigenvalue the optimization problem is no more differentiable and the present approach is not valid. This non-differentiability results in small oscillations which, even if they are invisible in Figure 5.21, prevent the stopping criteria based on maximum variation to be activated. This explains why the stopping criterion used in previous section has been replaced here by a criterion based on the number of iterations. The next section presents modifications of the optimization procedure required by the existence of repeated eigenvalues. The effects of those modifications are illustrated on basis of the same numerical benchmark.

5.6 Optimization with repeated eigenvalues

5.6.1 General framework

When an optimization problem includes a criterion over eigenvalues, it is very important to watch the multiplicity of the considered eigenvalues. Indeed, Seyranian *et al.* [96] show that the classical method used to optimize simple eigenfrequency leads to erroneous results if some eigenvalues coalesce and become multiple. Actually, when repeated eigenvalues exist, the optimization problem is non-differentiable and the sensitivity approach classically used for simple eigenvalues is not valid anymore.

The case of multiple eigenvalues is considered several times in literature (see for instance Refs. [53, 96, 123]). The common approach to compute the eigenvalues sensitivity is based on a perturbation technique as proposed by Courant and Hilbert [29] and is recalled in what follows. Let us first consider the following generic eigenproblem:

$$(\mathbf{A} - \lambda \mathbf{M})\,\mathbf{p} = \mathbf{0}, \tag{5.31}$$

where $\mathbf{A}$ and $\mathbf{M}$ are symmetric square matrices with n lines and columns. Additionally, we assume, without loss of generality, that the first m eigenvalues are repeated while the following $n - m$ are simple:

$$\lambda_1 = \lambda_2 = \ldots = \lambda_m < \lambda_{m+1} < \ldots < \lambda_n.$$

The perturbation method consists in applying a slight modification to the design parameters vector $\boldsymbol{\mu}$ along an arbitrary direction $\mathbf{e}$ with magnitude ε:

$$\boldsymbol{\mu} = \boldsymbol{\mu} + \varepsilon \mathbf{e} \text{ with } \|\mathbf{e}\| = 1.$$

As matrices $\mathbf{A}$ and $\mathbf{M}$ are supposed to evolve smoothly with the design parameters, their values after the perturbation can be estimated as follows:

$$\mathbf{A}(\boldsymbol{\mu} + \varepsilon \mathbf{e}) = \mathbf{A}(\boldsymbol{\mu}) + \varepsilon \sum_{i=1}^{N} \frac{\partial \mathbf{A}}{\partial \mu_i} e_i + o(\varepsilon), \tag{5.32}$$

$$\mathbf{M}(\boldsymbol{\mu} + \varepsilon \mathbf{e}) = \mathbf{M}(\boldsymbol{\mu}) + \varepsilon \sum_{i=1}^{N} \frac{\partial \mathbf{M}}{\partial \mu_i} e_i + o(\varepsilon). \tag{5.33}$$

Moreover, the perturbation also affects the eigenvalues from Eq. (5.31). The modification of the initially repeated m first eigenvalues can be written:

$$\lambda_j(\boldsymbol{\mu} + \varepsilon \mathbf{e}) = \tilde{\lambda} + \varepsilon \kappa_j(\boldsymbol{\mu}, \mathbf{e}) + o(\varepsilon), \qquad j = 1, \ldots, m, \tag{5.34}$$

with $\tilde{\lambda}$ equal to the repeated eigenvalue in the unperturbed state. κ_j may be interpreted as the directional derivative of λ_j in direction $\mathbf{e}$.

Conversely, the eigenvectors associated with the m repeated eigenvalues cannot be assumed to evolve continuously with the perturbation. Indeed, when there are repeated eigenvalues, the associated eigenvectors are determined up to an orthogonal transformation (see Ref. [29]). Therefore, we have to define another basis of the unperturbed eigenvectors:

$$\tilde{\mathbf{p}}_j = \sum_{k=1}^{m} \beta_{jk} \mathbf{p}_k, \qquad j = 1, \ldots, m. \tag{5.35}$$

Because eigenvectors of repeated eigenvalues are defined up to an orthogonal transformation, the $\tilde{\mathbf{p}}_j$'s are still eigenvectors from problem Eq. (5.31). The β_{jk} coefficients are unknown but have to be chosen such that the $\tilde{\mathbf{p}}_j$ eigenvectors evolve continuously with perturbation $\varepsilon \mathbf{e}$. The perturbed eigenvectors $\mathbf{p}_j(\boldsymbol{\mu} + \varepsilon \mathbf{e})$ can then be expressed using this new basis.

$$\mathbf{p}_j(\boldsymbol{\mu} + \varepsilon \mathbf{e}) = \tilde{\mathbf{p}}_j(\mu) + \varepsilon \boldsymbol{\nu}_j(\boldsymbol{\mu}, \mathbf{e}) + o(\varepsilon), \qquad j = 1, \ldots, m, \tag{5.36}$$

where appears $\boldsymbol{\nu}_j$: the eigenvectors directional derivative.

The next step consists in introducing perturbations (5.32), (5.33), (5.34) and (5.36) into the original eigenproblem Eq. (5.31). By neglecting high order terms in ε we get the following expression:

$$\sum_{i=1}^{N} \left(\frac{\partial \mathbf{A}}{\partial \mu_i} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_i} \right) \tilde{\mathbf{p}}_j e_i + \left(\mathbf{A} - \tilde{\lambda} \mathbf{M} \right) \boldsymbol{\nu}_j = \kappa_j \mathbf{M} \tilde{\mathbf{p}}_j.$$

We premultiply this equation by $\mathbf{p}_s^T$, which is any eigenvector associated with $\tilde{\lambda}$, one gets:

$$\sum_{i=1}^{N} \mathbf{p}_s^T \left(\frac{\partial \mathbf{A}}{\partial \mu_i} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_i} \right) \tilde{\mathbf{p}}_j e_i = \kappa_j \mathbf{p}_s^T \mathbf{M} \tilde{\mathbf{p}}_j, \qquad s = 1, ..., m,$$

as

$$\mathbf{p}_s^T \left(\mathbf{A} - \tilde{\lambda} \mathbf{M} \right) \boldsymbol{\nu}_j = \mathbf{0}.$$

Then, by inserting the definition of $\tilde{\mathbf{p}}_j$ (5.35), we get a system of m linear equations in terms of unknown coefficients β_{jk}:

$$\sum_{k=1}^{m} \beta_{jk} \left[\sum_{i=1}^{N} \mathbf{p}_s^T \left(\frac{\partial \mathbf{A}}{\partial \mu_i} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_i} \right) \mathbf{p}_k e_i - \kappa_j \delta_{sk} \right] = \mathbf{0}, \qquad s = 1, ..., m,$$

since $\mathbf{p}_s^T \mathbf{M} \mathbf{p}_k = \delta_{sk}$. Provided the values of the κ_j, this system of equations allows computing the m β_{jk} associated to $\tilde{\mathbf{p}}_j$. However, nontrivial solution of this system only exists if its determinant is equal to zero.

$$\det \left[\sum_{i=1}^{N} \mathbf{p}_s^T \left(\frac{\partial \mathbf{A}}{\partial \mu_i} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_i} \right) \mathbf{p}_k e_i - \kappa_j \delta_{sk} \right] = 0, \qquad k, s = 1, ..., m. \tag{5.37}$$

This equation allows us computing the coefficients κ_j $(j = 1, ..., m)$ as the solutions of a secondary eigenvalue problem of size m. By definition, these coefficients are the sensitivities of the multiple eigenvalue $\tilde{\lambda}$ for a perturbation along direction $\mathbf{e}$. Therefore, the later eigenproblem provides a way to compute the directional derivatives of $\tilde{\lambda}$ in a given $\mathbf{e}$ direction.

To simplify the notations, the so called *generalized gradient vectors* $\mathbf{f}_{sk}$ are defined by Seyranian *et al.* [96] as follows:

$$\mathbf{f}_{sk} = \left(\mathbf{p}_s^T \left(\frac{\partial \mathbf{A}}{\partial \mu_1} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_1} \right) \mathbf{p}_k, ..., \mathbf{p}_s^T \left(\frac{\partial \mathbf{A}}{\partial \mu_N} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \mu_N} \right) \mathbf{p}_k \right).$$

Notice that $\mathbf{f}_{sk}$ are vectors despite the two indexes. Moreover, because of the symmetry of matrices $\mathbf{A}$ and $\mathbf{M}$, we have $\mathbf{f}_{sk} = \mathbf{f}_{ks}$. Expression (5.37) is then shortened:

$$\det \left[\mathbf{f}_{sk} \mathbf{e} - \kappa_j \delta_{sk} \right] = 0, \qquad k, s = 1, ..., m. \tag{5.38}$$

5.6.2 Optimization methods for repeated eigenvalues

When dealing with non-differentiable functions in general optimization problems, one usually needs to resort to subgradient optimization methods. For a concave function f, the subgradient is defined as one direction $\mathbf{g}$ such that for any $\Delta \mathbf{x}$:

$$f(\mathbf{x} + \Delta \mathbf{x}) \leqslant f(\mathbf{x}) + \mathbf{g}^T \Delta \mathbf{x}. \tag{5.39}$$

The subdifferential is the set of all directions $\mathbf{g}$ that respect Eq. (5.39). If the function is differentiable in $\mathbf{x}$, the subdifferential contains one unique vector equal to the gradient

of the function. Conversely, in the case of a non-differentiable function, all directions included into the subdifferential are not necessarily ascent directions.

The basic subgradient optimization method consists in selecting one subgradient of the function and to modify design along this direction with step size α_k and restart the same procedure from the new design point. Even if the selected direction is not always an ascent direction, it is possible to prove for a concave function f that for a sufficiently small step size α_k, the new design point is closer from the optimum (see Ref. [11, Chap. 8]). Of course, the choice of the α_k sequence is of the utmost importance as shown in Ref. [11, Chap. 8] and Ref. [21, Chap. 6]. Alternatively, generalizations of the steepest ascent method to non-differentiable problems are also described by Bertsekas [21, Chap. 6]. The principle of this method is to select within the subdifferential the steepest ascent subgradient rather than selecting it arbitrarily.

However, in the case of repeated eigenvalues, alternative procedures have been developed. These methods take advantage of peculiarities of the problem and allow treating non-differentiability without having to resort to subgradient optimization methods.

To describe existing optimization methods for repeated eigenvalues, let us consider without loss of generality a problem where the two smallest eigenvalues are equal while the $n-2$ next eigenvalues are simple:

$$\lambda_1 = \lambda_2 < \lambda_3 < ... < \lambda_n. \tag{5.40}$$

The first eigenvalue is then repeated with multiplicity 2. As described in the previous section, for a given variation of the design variables $\Delta\boldsymbol{\mu}$, the corresponding increments of λ_1 and λ_2 are given by the solution of the eigenproblem:

$$\det \begin{bmatrix} \mathbf{f}_{11}\Delta\boldsymbol{\mu} - \Delta\lambda_1 & \mathbf{f}_{12}\Delta\boldsymbol{\mu} \\ \mathbf{f}_{21}\Delta\boldsymbol{\mu} & \mathbf{f}_{22}\Delta\boldsymbol{\mu} - \Delta\lambda_2 \end{bmatrix} = 0, \tag{5.41}$$

which is obtained by substituting $\Delta\boldsymbol{\mu}$ to $\varepsilon\mathbf{e}$ and $\Delta\lambda_j$ to $\varepsilon\kappa_j$ in Eq. (5.38).

Nevertheless, in an optimization context it is the inverse problem that needs to be solved; a design variable modification $\Delta\boldsymbol{\mu}$ has to be determined such that it improves the design (i.e. such that it leads to satisfactory $\Delta\lambda_i$). For this purpose, two methods can be found in literature. The first one proposed by Seyranian *et al.* [96] and recently used by Xia *et al.* [123] consists in solving an auxiliary linear system to determine an ascent direction. The second method has been proposed later by Krog *et al.* [53] and tackles the existence of repeated eigenvalue by adding equality constraints to the optimization problem. These two procedures are presented in the next sections.

Feasible ascent increment

The approach proposed by Seyranian *et al.* [96] consists in finding an update of the design variables $\Delta\boldsymbol{\mu}$ that both satisfies design constraints and improves eigenvalues by

solving an auxiliary linear system. Considering the example presented here above where the first eigenvalue is double (see Eq. (5.40)), let us suppose that we want to maximize the lowest eigenvalue of our system. Consequently, the resulting $\Delta\lambda_1$ and $\Delta\lambda_2$ have to be both positive. Considering Eq. (5.41), there are probably several choices of $\Delta\boldsymbol{\mu}$ that provide such result. Therefore, we can simplify the eigenproblem by imposing a restriction on $\Delta\boldsymbol{\mu}$ that results in a diagonal matrix in Eq. (5.41):

$$\mathbf{f}_{12}\Delta\boldsymbol{\mu} = 0. \tag{5.42}$$

With a diagonal matrix, the eigenproblem becomes straightforward and we have:

$$\Delta\lambda_1 = \mathbf{f}_{11}\Delta\boldsymbol{\mu},$$
$$\Delta\lambda_2 = \mathbf{f}_{22}\Delta\boldsymbol{\mu}.$$

Therefore Seyranian *et al.* impose the two following additional constraint on $\Delta\boldsymbol{\mu}$:

$$\Delta\lambda_1 = \mathbf{f}_{11}\Delta\boldsymbol{\mu} = 1, \tag{5.43}$$
$$\Delta\lambda_2 = \mathbf{f}_{22}\Delta\boldsymbol{\mu} = 1 - (\lambda_2 - \lambda_1)\,. \tag{5.44}$$

These two constraints both result in increasing λ_1 and λ_2. As, numerically speaking, an eigenvalue may be considered as multiple even if the computed eigenvalues are slightly different, the increment of λ_2 is reduced by the difference between the two eigenvalues. This strategy tries to increase both eigenvalues at the same time and tends to keep them equal to each other.

To find a design variables increment $\Delta\boldsymbol{\mu}$ that satisfies requirements formulated in equations (5.42), (5.43) and (5.44), Seyranian *et al.* propose to express $\Delta\boldsymbol{\mu}$ as a linear combination of the generalized gradient vectors,

$$\Delta\boldsymbol{\mu} = \gamma_{11}\mathbf{f}_{11} + \gamma_{12}\mathbf{f}_{12} + \gamma_{22}\mathbf{f}_{22}. \tag{5.45}$$

Inserting this last expression in the three constraints leads to the following system of equations that allows computing the values of γ_{11}, γ_{12} and γ_{22} and, by the way, defines an acceptable value of $\Delta\boldsymbol{\mu}$.

$$\begin{bmatrix} \mathbf{f}_{11}^T\mathbf{f}_{11} & \mathbf{f}_{11}^T\mathbf{f}_{22} & \mathbf{f}_{11}^T\mathbf{f}_{12} \\ & \mathbf{f}_{22}^T\mathbf{f}_{22} & \mathbf{f}_{22}^T\mathbf{f}_{12} \\ sym. & & \mathbf{f}_{12}^T\mathbf{f}_{12} \end{bmatrix} \begin{bmatrix} \gamma_{11} \\ \gamma_{22} \\ \gamma_{12} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 - (\lambda_2 - \lambda_1) \\ 0 \end{bmatrix}. \tag{5.46}$$

Solution of system (5.46) provides an ascent increment $\Delta\boldsymbol{\mu}$. However, as optimization problems generally include constraints, it is not possible to use this increment as the gradient of the objective function and resort to classical gradient based optimization. This is illustrated in Figure 5.22, which represents a two variables linear maximization problem with the isolines of the objective function f, the constraints and the current design (black point). In this example, the design variables update provided by a gradient based optimizer like *Conlin* differs from the gradient direction. Indeed, there is a better solution at the intersection of the constraints which can be reached by the optimizer.

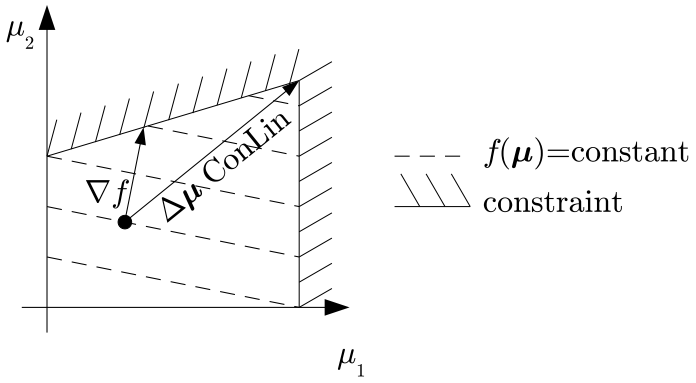


Figure 5.22: Example of configuration where *Conlin* will not follow the gradient.

In case of multiple eigenvalues, this approach would be problematic. Since the problem is non-differentiable, if the optimizer moves away from the ascent direction computed using (5.46), it is not possible to guarantee that the new direction is still an ascent direction. Therefore, when repeated eigenvalues are detected, gradient based optimizer must be disabled and replaced by a simpler procedure which consists in a line search along the ascent direction $\Delta\boldsymbol{\mu}$.

Additionally, constraints can be considered during the ascent direction computations (5.46) in order to find a design variable increment that both improves the eigenvalue and satisfies the design constraints. As shown by Seyranian, an optimization constraint can be taken into account by adding one term to the linear development of $\Delta\boldsymbol{\mu}$ (Eq. (5.45)) and one equation to the system Eq. (5.46). This additional term, equal to the gradient of the constraint multiplied by an additional unknown γ_i, ensures that $\Delta\boldsymbol{\mu}$ satisfies exactly the constraint. For inequality constraint, an active set strategy has then to be implemented to add or remove the constraints from the linear system when they become active or inactive.

Restriction of the design space

The main drawback of the previous method is that eigenvalues increments $\Delta\lambda_1$ and $\Delta\lambda_2$ have to be fixed a priori. With the increments definition (see Eq. (5.43) and Eq. (5.44)) proposed by Seyranian, the design modifications are computed in such a way that coalescent eigenvalues remain equal afterwards. This supposes that there are repeated eigenvalues at the optimum point and prevents repeated eigenvalues from separating during the optimization process.

Krog *et al.* [53] have developed an alternative approach. To explain this alternative method, let us consider the same eigenproblem as in previous section with the two first eigenvalues equal and the $n-2$ following simple. Their approach is based on the observation that when it is imposed that:

$$\mathbf{f}_{12}\Delta\boldsymbol{\mu} = 0. \tag{5.47}$$

The first order approximation of the two first eigenvalues increments with respect to design modifications are given by the same equations as if they were simple. Indeed, under condition Eq. (5.47) we have from the previous section:

$$\Delta\lambda_1 = \mathbf{f}_{11}\Delta\boldsymbol{\mu},$$
$$\Delta\lambda_2 = \mathbf{f}_{22}\Delta\boldsymbol{\mu},$$

where $\mathbf{f}_{11}$ and $\mathbf{f}_{22}$ are given by:

$$\mathbf{f}_{11} = \mathbf{p}_1^T \left(\frac{\partial \mathbf{A}}{\partial \boldsymbol{\mu}} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \boldsymbol{\mu}} \right) \mathbf{p}_1, \qquad \mathbf{f}_{22} = \mathbf{p}_2^T \left(\frac{\partial \mathbf{A}}{\partial \boldsymbol{\mu}} - \tilde{\lambda} \frac{\partial \mathbf{M}}{\partial \boldsymbol{\mu}} \right) \mathbf{p}_2,$$

which are equivalent to the expressions of the gradients of the two first eigenvalues if they were simple (see Eq. (5.19)).

Therefore, it means that gradient based optimizer can still be used when repeated eigenvalues exist provided that an equality constraint is added to the optimization problem. This additional constraint corresponds to the restriction of the design space by equation Eq. (5.47).

5.6.3 Implementation of the optimization procedure

The optimization procedure developed for repeated eigenvalues is based on the one used previously for simple eigenvalues. Of course, additional steps have been introduced in order to treat repeated eigenvalues correctly. A flowchart of the optimization procedure is presented in Figure 5.23.

Repeated eigenvalues handling

The first modification aims at detecting repeated eigenvalues. Once the eigenproblem is solved, the relative difference between consecutive eigenvalues is computed. As numerically, repeated eigenvalues will never be exactly equal, eigenvalue i is considered as repeated if its relative difference with respect to the next eigenvalue is lower than a predefined tolerance (set here to 1 %):

$$\text{abs}\left(\frac{\delta_i - \delta_{i+1}}{\delta_i} \right) < tol.$$

During this process, multiplicity of the eigenvalues is also computed by counting the number of consecutive eigenvalues respecting this condition.

Among methods described above to avoid issues related to non-differentiability of the optimization problem, both methods by Seyranian [96] and Krog [53] have been implemented. Comparison of the results provided by each method on the Clamped-Clamped microbridge application shows that they lead to similar results. Nevertheless, in the considered example, the feasible ascent method (Seyranian) takes more iterations (about

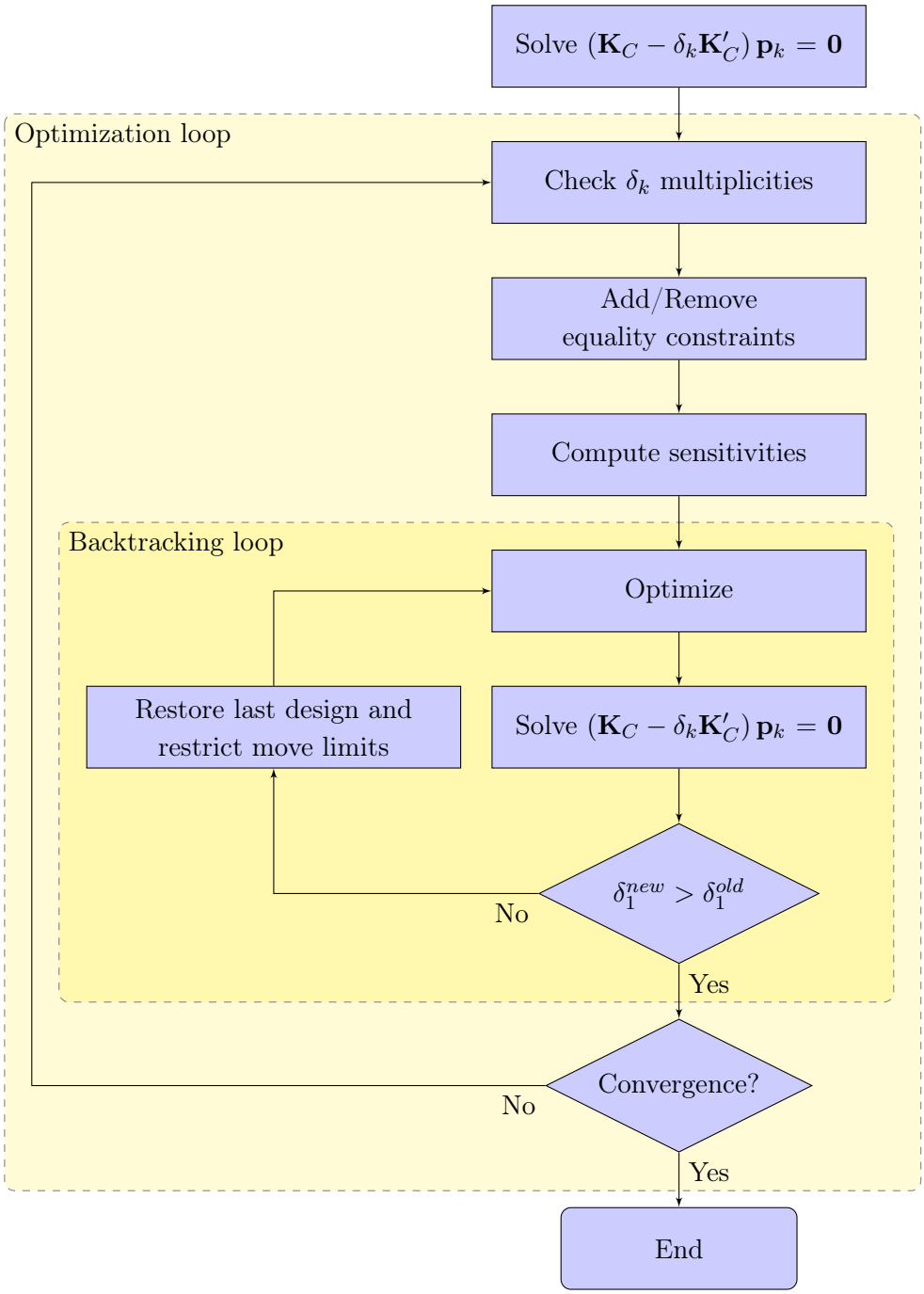


Figure 5.23: Implemented optimization procedure for repeated eigenvalues.

50% more) than Krog's method to reach the similar results. The method consisting in adding equality constraints (Krog) has the advantage to be more general than the other one as it does not involve any assumption about the evolution of repeated eigenvalues. Therefore, only results obtained using this method are presented.

Backtracking strategy

Moreover, a backtracking strategy based on Armijo condition (see Ref. [74, sec. 3.1]) has been added to the optimization loop. As presented in Figure 5.23, on every optimization iteration, the efficiency of design update provided by the optimizer is evaluated by computing the value of the lowest eigenvalue δ_1^{new}. If this value is greater (improved) than the one obtained with the last design δ_1^{old}, the optimization update is accepted. Conversely, if the first eigenvalue has decreased, the design update is rejected, the move limits are reduced and the optimizer executed once again from previous design. This backtracking procedure aims at avoiding oscillations of the optimization process. Application of the same strategy to simple eigenvalue optimization problem has also been attempted. However, inherent oscillatory behavior of the optimization leads to unreasonably slow process and failure to converge within 200 iterations (for the stopping criterion, only accepted iterations are counted).

5.6.4 Numerical application to the microbridge problem

To apply the repeated eigenvalue optimization to the Clamped-Clamped microbridge optimization problem, only the optimization procedure is modified while the optimization problem remains the same:

$$\begin{aligned} &\max_{\boldsymbol{\mu}} \left(\min_k \delta_k \right), \\ s.t. &\begin{cases} v\left(\boldsymbol{\mu}\right) \leqslant \overline{v}, \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i, \end{cases} \\ with &\left(\mathbf{K}_C - \delta_k \mathbf{K}'_C\right) \mathbf{p}_k = \mathbf{0}. \end{aligned}$$

To be consistent with the previous numerical example the same convergence criteria has been kept, the optimization process is stopped after 200 iterations. However, with the present optimization procedure, the optimization history is smoother and the final topology is presented in Figure 5.24 is clearly defined and converged structure. Notice that the final topology is similar to the one obtained with simple eigenvalues optimization but is more readable and free from undefined gray areas. Compared to the half-domain optimization we can see here that the optimizer increases the number of links between the arch and the imposed bridge.

The final results are also improved quantitatively as the final pull-in voltage reaches here 1054 V which is 6.7% better than simple eigenvalue optimization. Optimization

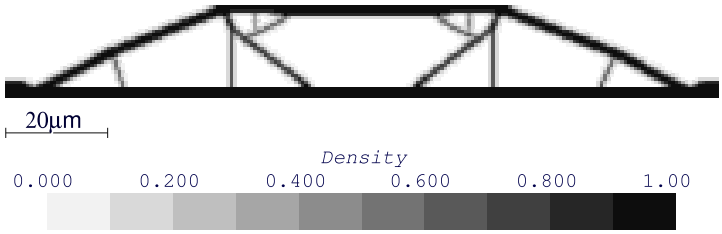
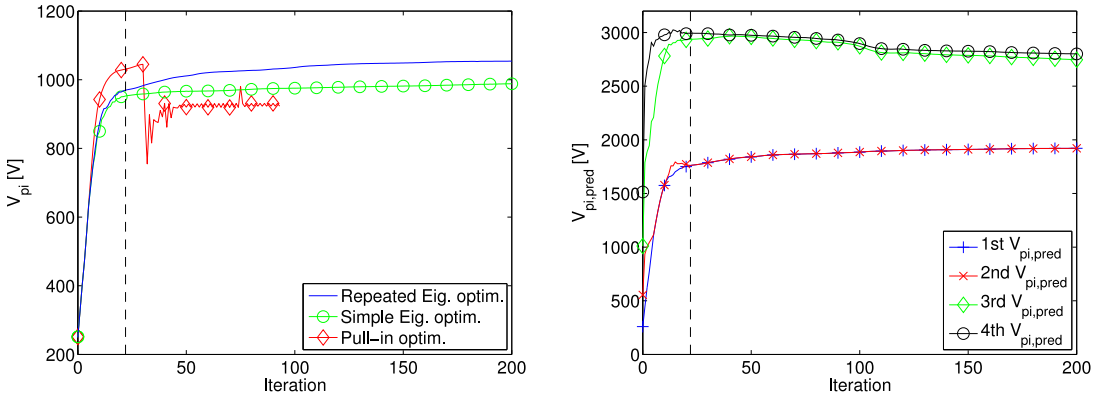


Figure 5.24: Resulting topology with repeated eigenvalues optimization.

history is plotted in Figure 5.25 from first iteration to iteration 200. The comparison of the present approach with the previously tested methods is presented in Figure 5.25(a) showing the history of the actual pull-in voltage across iterations while Figure 5.25(b) presents the predicted instability voltages. Repeated eigenvalue optimization procedure (i.e. addition of an equality constraint) is briefly activated between iterations 7 and 11 and definitely reactivated from iteration 22 (which is marked by the vertical dashed line). This can be verified in Figure 5.25(b) where the first two predicted pull-in voltages coalesce and separate at the corresponding iterations. Conversely to present results, the separation of the two first eigenvalues at iteration 11 is not observed if the optimization problem is solved with the feasible ascent method (Seyranian *et al.* [96]).



(a) Actual pull-in voltage.

(b) Predicted pull-in voltage.

Figure 5.25: Actual and predicted pull-in voltage history with stability eigenproblem optimization and repeated eigenvalues handling.

In Figure 5.25(a), the last results are compared to pull-in voltage optimization and simple eigenvalue optimization (without repeated eigenvalue treatment). At the beginning of the optimization process the results are identical but become different as soon as repeated eigenvalues are detected. Then, on the right of the dashed line, while the simple eigenvalue method stops increasing, the repeated optimization procedure keeps on improving the solution before stabilizing at 1054 V.

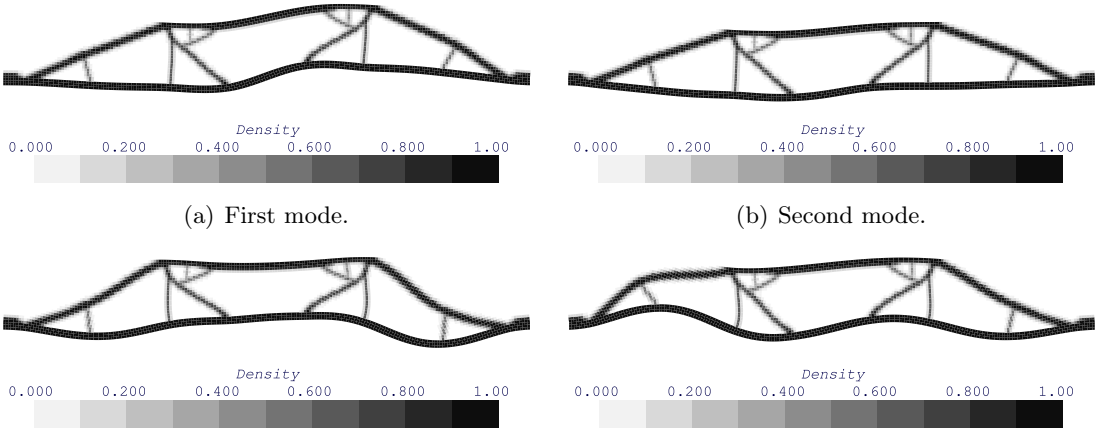
On the point of view of the predicted pull-in voltage presented in Figure 5.25(b) conclusions are the same as with simple eigenvalue optimization. The history of the lowest

predicted pull-in voltage is very similar to the one of the pull-in voltage. After definitive activation of the repeated eigenvalue strategy, the two first predicted pull-in voltage curves continue to increase and remain superposed. Numeric values presented in Table 5.8 show that the first two eigenvalues are even closer from each other at the end of the optimization process than in Table 5.7.

Order	Value [V^2]	Predicted V_{pi} [V]
1	3.62968e+006	1920.9
2	3.62969e+006	1920.9
3	7.48028e+006	2746.0
4	7.78511e+006	2801.0

Table 5.8: Final first four eigenvalues with repeated eigenvalues optimization.

As an illustration, Figure 5.26(a) and Figure 5.26(b) present the two eigenmodes corresponding to the repeated eigenvalue. The first mode corresponds to a collapse of one half of the microbeam while the second one shows a global collapse of the structure. These modes are orthogonalized in matrix $\mathbf{K}'_C$ metric. For illustrative purpose, third and fourth modes are presented in Figures 5.26(c) and 5.26(d). These last modes correspond to more localized instabilities but are purely 'fictitious' since the associated eigenvalues are larger than the two first ones.



(a) First mode. (b) Second mode.

(c) Thrid mode. (d) Fourth mode.

Figure 5.26: Mode shapes corresponding to the first four eigenvalues.

5.7 Sensitivity validation

5.7.1 Comparison with pull-in voltage sensitivities

The results of the last numerical applications (see Figure 5.25) confirm (at least for the present example) that the eigenproblem formulation provides design variables increments which lead to improved designs as the actual pull-in voltage is increased. To

complement this qualitative observation the sensitivities provided by the eigenproblem approximation are compared in this subsection with the actual pull-in voltage sensitivities computed by Eq. (5.29).

To evaluate the accuracy of the sensitivities, they are compared for the initial and for the optimal material distribution from Section 5.6.4. Indeed, these two distributions correspond to very different situations. At the first iteration, the evaluation voltage has been chosen very close to the actual pull-in voltage in order to maximize the accuracy of the prediction. Conversely, as the evaluation voltage is fixed along the optimization process, it becomes several times smaller than the final pull-in voltage. Moreover, at the last iteration the first eigenvalue is repeated while it is simple at the beginning.

As sensitivity vectors have a large number of elements, two global indicators are used to evaluate the agreement between the i^{th} predicted pull-in voltage sensitivities $\boldsymbol{\nabla} V_{pi,pred,i}$ with the actual pull-in sensitivities $\boldsymbol{\nabla} V_{pi}$:

- L_2 norm ratio:

$$r_i = \frac{\|\boldsymbol{\nabla} V_{pi,pred,i}\|}{\|\boldsymbol{\nabla} V_{pi}\|}.$$

- The angle between the two vectors:

$$\theta_i = \cos^{-1}\left(\frac{\boldsymbol{\nabla} V_{pi,pred,i} \cdot \boldsymbol{\nabla} V_{pi}}{\|\boldsymbol{\nabla} V_{pi,pred,i}\| \, \|\boldsymbol{\nabla} V_{pi}\|}\right).$$

These two indicators allow easier comparison of the sensitivity vectors.

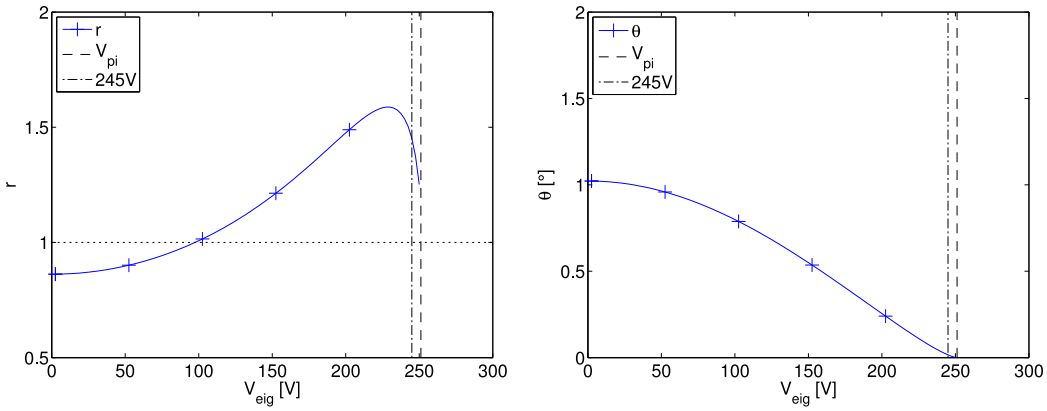
Initial design

At first, the initial material distribution is considered as the basis for comparison. For this fixed design, the sensitivities of the first predicted pull-in voltage $\boldsymbol{\nabla} V_{pi,pred,1}$ are computed for $V_{eig} = 245$ V (as in Sections 5.5.3 and 5.6.4) and compared to the sensitivities of the actual pull-in voltage $\boldsymbol{\nabla} V_{pi}$. It comes out that the two gradients are almost parallel as the angular distance is equal to 0.02°. But a significant difference can be noticed concerning the magnitude of the two vectors as the ratio is equal to 1.45.

To explain the magnitude difference between the two gradients, one has to remind that the stability eigenproblem only provides an estimate of the pull-in voltage. Moreover, the accuracy of this prediction decreases if the eigenproblem is formulated far from the instability point. Then, a design perturbation that results in a pull-in voltage increase also reduces the accuracy of the next prediction since the eigenproblem is formulated at a fixed voltage level. In other words, if the predicted pull-in voltage is overestimated, the design perturbation makes the error worse as the prediction point is farther from the new instability point.

Even if they do not possess the same magnitude, the two sensitivity vectors are oriented towards the same direction which is the most important in our optimization context where pull-in voltage (or its approximation) appears as an objective function. However, the magnitude difference may be an issue if pull-in voltage is involved in a design constraint. The behavior of the optimization process should be watched closely in this case.

In order to determine the effect of the evaluation voltage, we can also compute r_1 and θ_1 for different values of the evaluation voltages V_{eig} (i.e. on several points of the equilibrium curve). The results of these computations are presented in Figures 5.27 against the evaluation voltage. In both Figures 5.27, the vertical dashed line represents the pull-in voltage level while the dot-dashed line represents the original evaluation voltage for the application of Section 5.6.4 (i.e. 245 V).



(a) Norm ratio. (b) Angle.

Figure 5.27: Variation of the sensitivities against V_{eig} at initial configuration.

Figure 5.27(a) shows that the norms of the two sensitivity vectors are almost always different. Nevertheless, in Figure 5.27(b) we can see that angular distance between the two gradients decreases monotonously with the evaluation voltage which confirms that the instability is more accurately captured when the eigenproblem is formulated close to pull-in.

Final design

As the optimization process goes on, pull-in voltage increases while the evaluation voltage remains fixed. Meanwhile, the two first instability eigenvalues have coalesced. Therefore, the previous discussion has been carried out on the final design. Since for the final design the first eigenvalue is double, the magnitude ratio and the angular distance are now represented for the two first predicted pull-in voltages (r_1 and θ_1 corresponding to the first predicted instability voltage and, r_2 and θ_2 to the second). The results are presented in Table 5.9.

i	r_i	θ_i [°]
1	1.39	54.5
2	1.09	29

Table 5.9: Sensitivity comparison for final design.

Conversely, to the initial design, the final design leads to estimated pull-in voltage sensitivities very different from the actual pull-in voltage sensitivities. Even if the norm ratios are relatively close to one for both modes, the two angular distances take very high values. Actually, as the first eigenvalue is repeated, the corresponding eigenmodes are defined up to a linear combination and can therefore give rise to sensitivities which are very different with respect to pull-in voltage sensitivities. At this point, the two approaches are not comparable since eigenproblem formulation attempts to improve the structure with respect to two instability modes while the pull-in voltage approach is only able to capture one of these instability modes.

However, it comes out that the relative difference between the first two eigenvalues, and by consequence, the multiplicity of the first eigenvalue are very sensitive to the evaluation voltage. As presented in Figure 5.28, the distance between the two eigenvalues is actually minimal for the evaluation voltage used for optimization indicated by the vertical dot-dashed line. Lower and higher values of the evaluation voltage lead to larger relative difference. For evaluation voltages greater than 609 V, the two eigenvalues cannot be considered as coalescent according to our criterion as the relative difference is larger than 1%. This difference reaches 60% if $V_{eig} = 1050$ V.

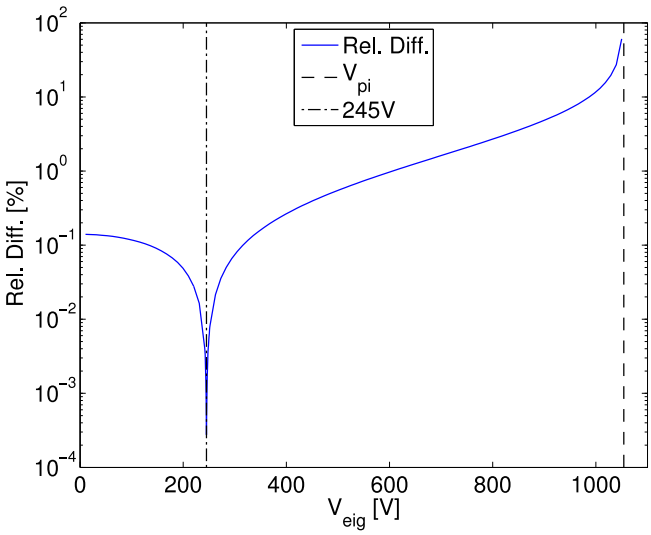
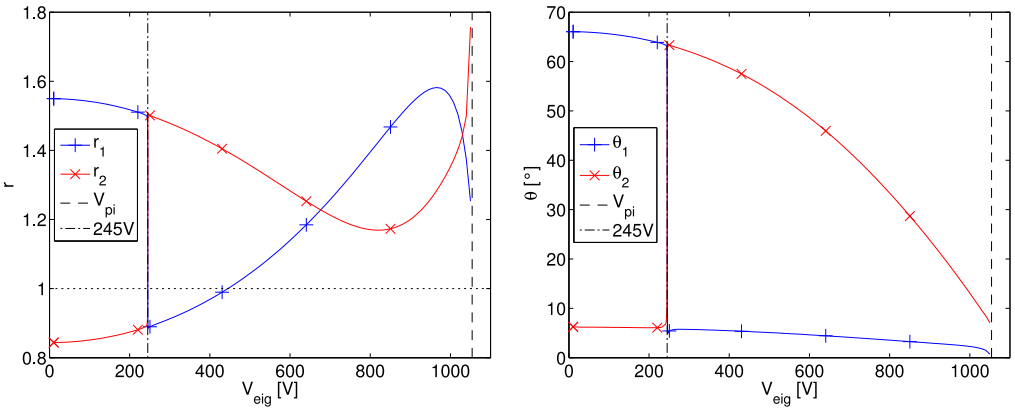


Figure 5.28: Relative difference between first and second eigenvalues against V_{eig} at final configuration.

Moreover, the graphs of r_1, r_2, θ_1 and θ_2 against the evaluation voltage in Figures 5.29 show that the eigenvalues also swap depending on the evaluation voltage. Indeed if

we consider the angular distance in Figure 5.29(b), for voltages lower than 245 V, the second predicted instability mode is closer to the actual instability mode. Inversely for higher values of the evaluation voltage it is the first instability mode which fits the best to the actual mode.



(a) Norm ratio. (b) Angle.

Figure 5.29: Variation of the sensitivities against V_{eig} at final configuration.

Nevertheless, there is always one of the first two modes gradient direction close to the pull-in voltage gradient. The angular distance is equal to 6° for lowest values of the evaluation voltage which is higher than for the initial configuration. However, it also converges towards 0° if evaluation voltage is increased. Concerning, the norms of the sensitivity vectors, as previously, they are almost always different from actual pull-in sensitivity. Therefore, these curves lead to similar conclusions as the one presented for the initial design.

Synthesis of the sensitivities comparison

In the considered numerical application, the combination of a fixed evaluation voltage with the increase of pull-in voltage during optimization process leads to a wrong evaluation of the multiplicity of the instability mode. The instability mode that looks double from nominal evaluation voltage appears to be simple when the eigenproblem is formulated closer from instability point. Additionally, deterioration of the sensitivities provided by the stability eigenproblem has been noticed when evaluation voltage is small with respect to pull-in voltage.

5.7.2 Modification of the optimization procedure

The solution to the issue raised in previous section is to adapt the evaluation voltage to the current design in order to keep it close to the instability point. This is the approach chosen by Lindgaard [59]. However, it must be stressed that modifying the

evaluation voltage during the optimization problem is not strictly rigorous. Indeed, the sensitivities of the stability eigenproblem have been derived under the assumption of a fixed evaluation voltage.

Nevertheless, we propose to keep the evaluation voltage proportional to the current pull-in voltage:

$$V_{eig} = \alpha V_{pi}. \tag{5.48}$$

This update scheme requires to first compute the pull-in voltage for every new design prior to formulate the stability eigenproblem.

To test the interest of using a variable V_{eig}, the update formula from Eq. (5.48) has been implemented in the stability eigenproblem optimization procedure with handling of repeated eigenvalues presented in Section 5.6. However, updating V_{eig} at every iteration slows down the optimization process as it perturbs the evolution of the eigenvalues which take much more iterations to coalesce. Therefore, we choose to update V_{eig} less often and to keep it constant for a defined number of iterations so that the repeated eigenvalue optimization procedure can be activated earlier.

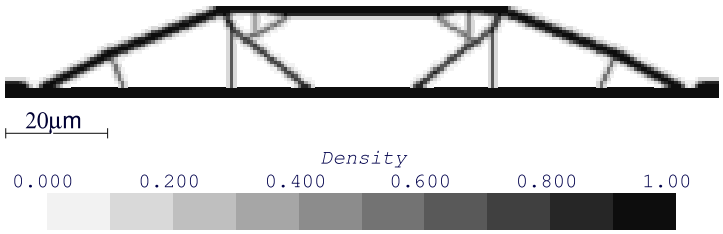
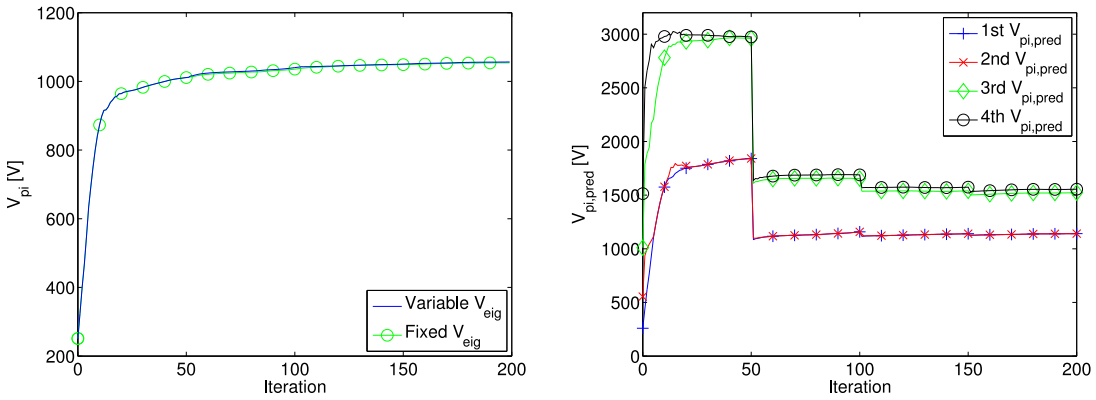


Figure 5.30: Resulting topology with repeated eigenvalues optimization.

The Clamped-Clamped microbridge optimization problem presented in previous sections has been treated once again. The coefficient α is chosen to 0.95 in order to avoid convergence problem close to pull-in point while staying as close as possible from this point. After a few trials, keeping V_{eig} constant for 50 iterations provided the best results. The topology obtained with these update parameters is presented in Figure 5.30. The structural topology is identical to the one obtained with fixed evaluation voltage (see Figure 5.24).

To compare quantitatively the efficiency of the two methods, Figure 5.31(a) plots the history of the actual pull-in voltage during the 200 iterations of the optimization process. The results obtained with the variable evaluation voltage are compared to the results obtained in Section 5.6.4 with a fixed voltage strategy. As we can see, the pull-in voltage history is almost the same. At the end of the optimization process, 1056 V are reached here against 1054 V previously, which means an increase of less than 0.2%.

The history of predicted pull-in voltage plotted in Figure 5.31(b) is much different with respect to fixed evaluation voltage optimization. The difference is mainly due to the better accuracy of the pull-in voltage prediction as it is now performed closer to instability. For instance, the final predicted instability voltage overestimates the actual



(a) Actual pull-in voltage.

(b) Predicted pull-in voltage.

Figure 5.31: Actual and predicted pull-in voltage history with stability eigenproblem optimization and repeated eigenvalues handling.

one by only 8% while, with a fixed evaluation voltage, the overestimation is of 82% (see Figure 5.25(b)). The first update of V_{eig} is clearly visible in the history of predicted pull-in voltage which drops at iteration 50. Subsequent updates have lesser impact since the modification of V_{eig} is smaller.

5.7.3 Summary

The comparison of the sensitivities provided by the stability eigenproblem to the pull-in voltage sensitivities has shown that the stability eigenproblem is a very good basis in order to optimize pull-in voltage. Indeed, as long as the evaluation voltage is close to pull-in point, the gradient vectors of both functions are almost parallel.

However, as pull-in voltage increases during optimization the quality of the eigenproblem sensitivities may be affected. Therefore, a modified optimization procedure such that the evaluation voltage follows the modifications of pull-in voltage has been implemented and tested. Nevertheless, the results provided by the modified optimization procedure do not differ significantly from the originals. This confirms (at least for the present numerical application) the validity of the sensitivities provided by the stability eigenproblem even with a fixed evaluation voltage.

5.8 Conclusion

As shown on the basis of a numerical application, pull-in voltage optimization procedure developed in Chapter 3 may suffer from oscillations that prevent from reaching solution. The analysis of these oscillations has shown that this issue is similar to the mode switching problem that arises in optimization problems involving an eigenvalue as objective function. The classical solution to this issue consists in including several

eigenvalues of the eigenproblem in a max-min formulation. However, as the pull-in voltage maximization optimization problem is not formulated as an eigenproblem, direct application of this solution is not possible. Therefore, the first part of this chapter has been dedicated to the development of an eigenproblem formulation for pull-in instability phenomenon.

The approach chosen to develop the eigenproblem formulation is inspired by the work of Lindgaard *et al.* [59]. However, conversely to the case treated by Lindgaard, it comes out that it is not possible to build an eigenproblem formulation on basis of the tangent stiffness matrix itself. Indeed such formulation leads, in the present framework, to complex eigenvalues that are not possible to interpret physically.

However, by using a condensed version of the tangent stiffness matrix, we show that it is possible to obtain a satisfactory eigenproblem formulation. This modified tangent matrix is computed by condensing electrostatic DOFs onto the mechanical part of the matrix. Then the eigenproblem is established using a *two-point* formulation. When this eigenproblem is formulated on a point of the equilibrium curve, its eigenvalues allow predicting upcoming instabilities and the associated pull-in voltages. The accuracy of the prediction improves as the eigenproblem is formulated closer to the instability point.

The established stability eigenproblem leads to the development of an alternative optimization procedure aiming at maximizing pull-in voltage. This new optimization method is based on a multiobjective formulation used to maximize the lowest eigenvalue. To reach good results, the ability to handle repeated eigenvalues is needed. Consequently, the method developed by Krog *et al.* [53] has been included. Thanks to the availability of semi-analytic sensitivities for the eigenvalues, the new optimization procedure is able to circumvent oscillation issues observed with pull-in voltage optimization. Moreover, numerical results show that even if the eigenproblem formulation is an approximation of the actual pull-in voltage optimization problem, eigenproblem formulation leads to improvement of pull-in voltage.

These results are confirmed by the comparison of the predicted pull-in voltage sensitivities with respect to the actual pull-in sensitivities. Indeed, the orientations of the two function gradients are almost identical when the eigenproblem is formulated close to pull-in point. Nevertheless, the quality of the eigenproblem sensitivities seems to deteriorate during optimization because pull-in voltage moves away from the fixed voltage at which the eigenproblem is formulated. Therefore a modified version of the optimization procedure is proposed in which the evaluation voltage follows the increase of pull-in voltage in order to preserve accuracy of the eigenproblem through the optimization process. No significant modifications could be noticed in numerical results which show the robustness of the original procedure.

Chapter 6

Generalization of pull-in voltage maximization

6.1 Introduction

Along the Chapters 4 and 5, a simplified version of the optimization problem has been investigated in order to develop a design function based on the pull-in voltage step by step. In the simplified problem, the optimization domain and the electric domain are separated by a perfectly conducting layer of material. Because of this assumption, the design domain is purely mechanical even if the analysis problem is still multiphysic.

In the present chapter, we propose to achieve one additional step by including the electric domain into the design domain. As a result, the optimization domain becomes multiphysic and the optimization process is able to modify the electrostatic pressure distribution. Therefore, the generality of the optimization procedure is increased. Nevertheless, the generalization of the optimization problem requires the modification of the electromechanical modeling such that design can modify both mechanical and electrostatic fields. The adaptations of the modeling procedure make the tangent stiffness matrix asymmetric such that it is not possible to apply the stability eigenproblem approach form Chapter 5. Conversely, the asymmetry of the tangent stiffness matrix is not an issue for pull-in design function. Thus, this chapter considers the generalization of the optimization problem considered in Chapter 4.

Along the development of the generalized optimization procedure, some aspects have been carefully considered. At first, alike mechanical properties, electrostatic material properties have to be functions of the pseudo-density distribution. The choice of the material model is discussed on the basis of theoretical bounds on multiphase material properties and a simple 1D example is used to illustrate the approach. Moreover, as electrostatic forces are now highly design dependent, we have to take care about the possibility of local instability related to the material model as reported in design dependent loads problem (see for instance Ref. [27]). The occurrence of local instability

is also investigated using the 1D example.

Moreover, unexpected issues have also been encountered. The origin of these difficulties is studied in the second part of this chapter. Once the source of the difficulties is understood, two solutions are proposed. The most efficient one is selected for the numerical applications.

6.2 Generalized optimization problem

6.2.1 Design problem description

The objective of the present chapter is to further develop the optimization procedure presented in Chapter 4. The pull-in maximization optimization problem studied in Chapter 4 is generalized by removing the assumptions that lead to a separation between mechanical optimization domain and electrostatic domain. The resulting optimization problem is now schematically represented in Figure 6.1.

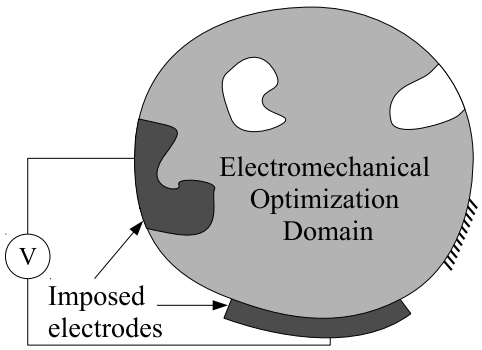


Figure 6.1: Schematic view of the generalized optimization problem.

Consequently, the electrostatic domain is now embedded into the optimization domain alike the mechanical domain. The shape and topology of both physical domains are determined by the material distribution ruled by the optimization procedure. Nevertheless, thanks to the monolithic finite element formulation we can avoid resorting to complicated modeling procedure as the one used by Raulli and Maute [86]. Indeed, the monolithic formulation allows modeling the optimization domain using only one finite element model as it has been done by Yoon and Sigmund [129]. The physical behavior of each element is determined by interpolating the material properties between void and design material properties. In order to reflect the effect of design on the shape of the electrostatic domain, the dielectric permittivity of each element needs to be computed according to its pseudo-density.

Another consequence of this modification is that the electrostatic forces application points are now design dependent. Design dependent forces tend to complicate the optimization problem as noticed by Bruyneel and Duysinx in [27] for instance in self weight problems.

In the perspective of building step by step an optimization procedure able to control pull-in voltage, the number of design functions is limited in this chapter to the minimum as in Chapter 4. Therefore, the objective function is again to maximize pull-in voltage with an upper bound on the available design material. Mathematically, this optimization problem can be written:

$$\max_{\boldsymbol{\mu}} V_{pi}\left(\boldsymbol{\mu}\right), \quad s.t. \begin{cases} v\left(\boldsymbol{\mu}\right) \leqslant \overline{v} \\ \mu_{min} \leqslant \mu_i \leqslant 1 \quad \forall i \end{cases}.$$

Even if the optimization problem has been generalized, the sensitivities of the pull-in voltage function can still be computed using the approach presented in Chapter 4 that leads to the general expression of the sensitivities in Eq. (4.2).

6.2.2 Electrostatic modeling modifications

In MEMS actuators, the structural parts may also be used as electric conductors in order to conduct the electric potential where electrostatic forces have to be generated. Unfortunately, electrostatics and electric steady conduction belongs to two different simplifications of Maxwell electromagnetic equations. In electrostatics, it is assumed that electric charges are fixed while they have to be free to move for electric conduction. Therefore, strictly speaking, electric steady conduction cannot be modeled using electrostatic finite element formulation.

However, since electrostatic equations and steady-state conduction equations are both ruled by a Poisson equation (see [32, Chap. 2]), a perfect conductor behavior can be obtained with electrostatic modeling. As suggested by Konrad [52], this is done by replacing the actual permittivity ε of the perfectly conducting part of the domain by a fictitious permittivity $\tilde{\varepsilon}$ that has a high value with respect to void permittivity (about $10^4 \cdot \varepsilon_0$). This technique is also used by Yoon and Sigmund [129], avoiding the development of a coupled formulation that takes into account three physical fields: electrostatics, electric conduction and mechanics.

Electrostatic potential computation

To illustrate the possibility to model perfectly conducting behavior using a fictitious permittivity, let us consider the 1D electrostatic example presented in Figure 6.2. The model represents a parallel plate capacitor where side effects are neglected. The first plate is located at node 0 and its voltage is set to 0 V. Node 2 corresponds to the second plate on which a voltage V is applied.

The model is divided in two elements by node 1, each of them having length s. The first finite element (on the left in Figure 6.2) is prescribed to be void so that its permittivity

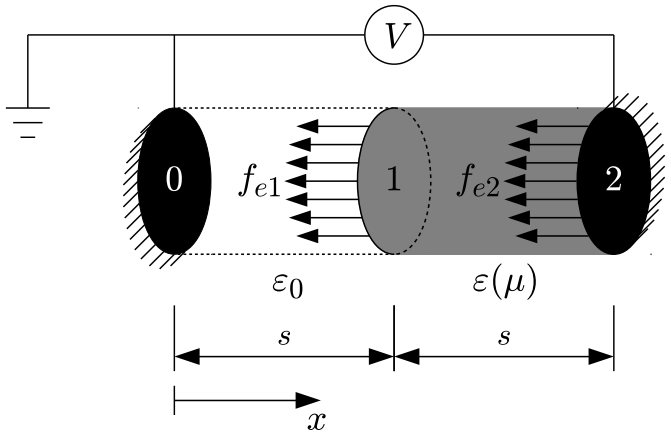


Figure 6.2: Sketch of the 1D electrostatic problem.

is equal to ε_0. The second element, which links node 1 to node 2, has a variable pseudo-density μ that modifies its physical behavior. When its pseudo-density is equal to zero, the element behaves as void. Conversely, if its pseudo-density is equal to one, it is supposed to be perfectly conducting. Therefore, its permittivity is defined as a function of the pseudo-density. The second element permittivity must be equal to ε_0 when the pseudo-density is zero and to a fictitious high value ε_{max} when the pseudo-density is high.

The first numerical experiment that we propose consists in fixing the pseudo-density μ to 1 so that $\varepsilon(\mu) = \varepsilon_{max}$ and in studying the effect of ε_{max} on the voltage at node 1: V_1. The objective here is to determinate the ratio between ε_{max} and ε_0 to render a perfectly conducting behavior for the right side element (i.e. $V_1 = V$). The voltage at node 1 can be computed as a function of the applied voltage using the fact that the electric displacement must be equal in both parts of the domain. After some basic algebra, we get the expression of V_1:

$$V_1 = \frac{\varepsilon_{max} V}{\varepsilon_{max} + \varepsilon_0}.$$

The value of V_1 as a function of the ratio $\varepsilon_{max}/\varepsilon_0$ is plotted in Figure 6.3 for $V = 1$ V. When the ratio $\varepsilon_{max}/\varepsilon_0$ is equal to 1, the two elements have the same behavior. Therefore, the voltage increases linearly over the domain along x direction and V_1 is equal to the half of V. Then as ε_{max} increases the curve shows that V_1 converges towards the applied voltage. According to Figure 6.3, a permittivity ratio of 10^3 is sufficient to render a conducting behavior over the solid part of the domain.

Computation of electrostatic forces

This first numerical experiment illustrates the possibility to model a perfect conducting behavior in electrostatic modeling by affecting a very high permittivity to the conducting parts. In their work on topology optimization of electrostatic actuators, Yoon and

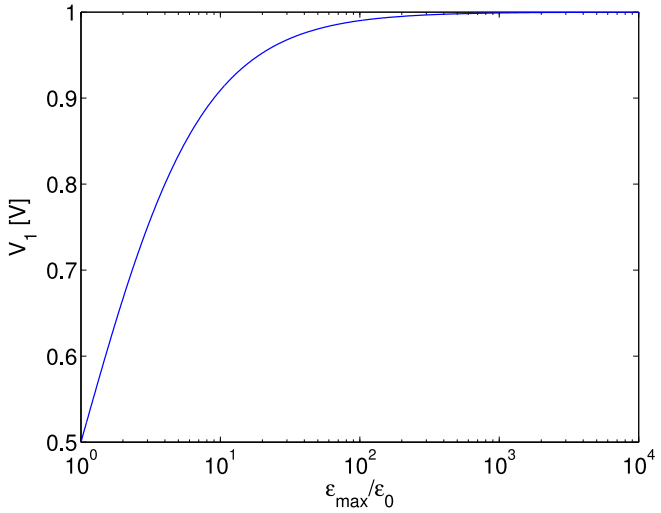


Figure 6.3: Voltage V_1 against maximal permittivity ε_{max}.

Sigmund [129] also resort to this method to model conducting behavior. However, in order to preserve the physical meaning of the model, they choose to define two distinct permittivities. The first one is the fictitious permittivity $\tilde{\varepsilon}$; it is used for voltage distribution computation and it reaches a high value for solid parts of the domain. The second permittivity ε is involved into the electrostatic forces computations; it varies between void permittivity ε_0 and the actual permittivity of the design material ε_s. Therefore, two interpolations of the material properties are needed for the electrostatic modeling. The interpolation schemes chosen by Yoon and Sigmund are SIMP laws with a parameter $p = 3$ for both properties:

$$\begin{aligned} \text{Force calculation:} \quad & \varepsilon(\mu) = \varepsilon_0 + \mu^p (\varepsilon_s - \varepsilon_0), \\ \text{Voltage distribution:} \quad & \tilde{\varepsilon}(\mu) = \varepsilon_0 + \mu^p (\tilde{\varepsilon}_{max} - \varepsilon_0) . \end{aligned} \tag{6.1}$$

For the sake of conciseness, $\varepsilon(\mu)$ and $\tilde{\varepsilon}(\mu)$ are respectively replaced by ε and $\tilde{\varepsilon}$ in the following.

Let's experiment numerically the effect of using two distinct dielectric constants on the value of the electrostatic forces. The simple model used for the previous experiment (see Figure 6.2) is used once again. Instead of fixing the density μ of the right part of the domain, the objective here is to study how the electrostatic force behaves with respect to variation of μ. First, let us express V_1 using the new notations:

$$V_1 = \frac{\tilde{\varepsilon} V_2 + \tilde{\varepsilon}_0 V_0}{\tilde{\varepsilon}_0 + \tilde{\varepsilon}} = \frac{\tilde{\varepsilon} V}{\tilde{\varepsilon}_0 + \tilde{\varepsilon}}, \tag{6.2}$$

where $\tilde{\varepsilon}_0$ denotes the fictitious permittivity of the void element. It is equal to ε_0 and it is here used to keep the notations as consistent as possible.

The electrostatic force that is applied on node 2 depends on the electrostatic field $\mathbf{E}_{12}$

between nodes 1 and 2 and on the permittivity of the media $\varepsilon\,(\mu)$:

$$f_{e2} = -\frac{\varepsilon}{2}\left\|\mathbf{E}_{12}\right\|^2 = -\frac{\varepsilon}{2}\frac{\left(V_2 - V_1\right)^2}{s^2}\,, \tag{6.3}$$

where $\mathbf{E}_{12}$ is replaced by the voltage gradient. In this expression, V_1 can be substituted using Eq. (6.2):

$$f_{e2} = -\frac{V^2}{2s^2}\frac{\tilde{\varepsilon}_0^2\varepsilon}{\left(\tilde{\varepsilon}_0 + \tilde{\varepsilon}\right)^2}\,, \tag{6.4}$$

which only depends on the applied voltage and on the material model.

Similarly, we can compute the electrostatic force acting on node 1. This node is submitted to two opposed electrostatic forces. The first one is due to the variable density element and has the same direction as the x axis. Conversely, the second one is directed in the opposite direction and is due to the void element. The sum of the two electrostatic forces gives:

$$f_{e1} = \frac{\varepsilon}{2}\frac{\left(V_2 - V_1\right)^2}{s^2} - \frac{\varepsilon_0}{2}\frac{\left(V_1 - V_0\right)^2}{s^2} = \frac{V^2}{2s^2}\frac{\tilde{\varepsilon}_0^2\varepsilon - \varepsilon_0\tilde{\varepsilon}^2}{\left(\tilde{\varepsilon}_0 + \tilde{\varepsilon}\right)^2}\,, \tag{6.5}$$

where again V_1 is substituted.

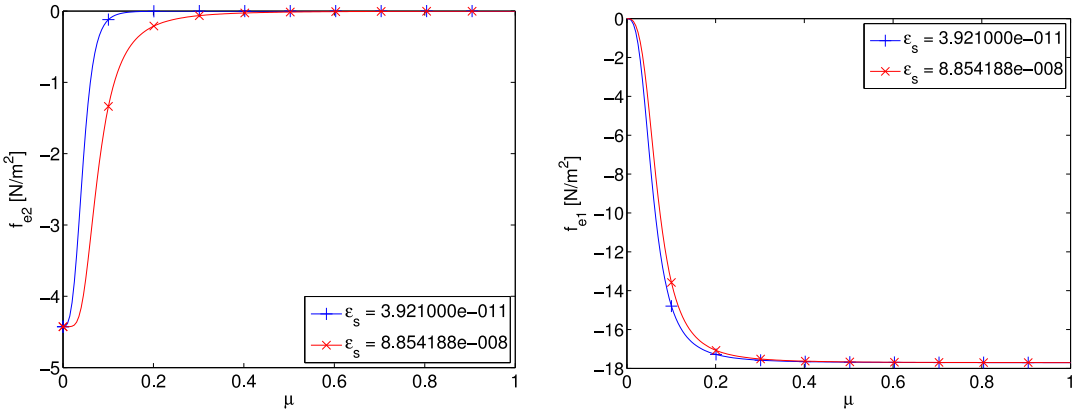
Finally, it is worth computing the resultant force that is applied to the clamping at node 2. The resultant is equal to the sum of f_{e1} and f_{e2}:

$$f_{eTot} = f_{e1} + f_{e2} = -\frac{V^2}{2s^2}\frac{\varepsilon_0\tilde{\varepsilon}^2}{\left(\tilde{\varepsilon}_0 + \tilde{\varepsilon}\right)^2} = -f_{e0}\,.$$

Moreover, it is obviously equal in magnitude to the electrostatic force acting on node 0.

The values of the three forces are plotted against the pseudo-density μ in Figure 6.4 for a fixed input voltage $V = 1$ V and an element size $s = 0.5$ µm. For each nodal force, two curves are drawn corresponding to different values of ε_s (i.e. the maximum value of ε in Eq. (6.1)) while the fictitious permittivity $\tilde{\varepsilon}$ interpolation remains the same. The first curve is obtained with $\varepsilon_s = 3.92 \cdot 10^{-11}$ F/m which is the actual dielectric constant value for silicon. This curve corresponds to the case where different permittivity values are used for electric potential distribution computation and for electrostatic forces evaluation. Conversely, the second curve is computed with $\varepsilon_s = \tilde{\varepsilon}_{max} = 10^4\varepsilon_0$, which means that the two permittivities are equal so that the fictitious permittivity value is used for electrostatic forces computation.

The electrostatic force f_{e2} on node 2 (Figure 6.4(a)) is the most sensitive to the permittivity modification. In both cases, when the second element is void (i.e. $\mu = 0$) a negative electrostatic force pulls node 2 in direction of node 0. However, when pseudo-density is increased, this electrostatic force goes to zero much faster if the actual permittivity of silicon is used than if the $\tilde{\varepsilon}_{max}$ is used for computation of electrostatic forces. This observation can be explained by noting that ε is in the numerator of Eq. (6.3) and Eq. (6.4) so that if ε is larger the electrostatic force magnitude is also larger. Nevertheless, the electric field in Eq. (6.3) is decreasing faster than ε is increasing and the electrostatic force vanishes in both cases.



(a) Electrostatic force on node 2.

(b) Electrostatic force on node 1.

Figure 6.4: Electrostatic forces in simple model in Figure 6.2 as functions of μ.

The relative influence of ε_s is smaller for f_{e1} as we can see in Figure 6.4(b). This can be expected since in Eq. (6.5), ε only appears in one of the terms that contribute to the electrostatic force. Moreover, the two curves only differ for a limited range of pseudo-density and they converge to the same value for $\mu = 0$ and $\mu = 1$.

If the two nodal electrostatic forces are influenced by a modification of the permittivity ε, it is not the case of their sum as shown in Figure 6.5. Actually, the differences noted previously compensate each other so that, the resultant force is independent from the choice of ε_s. The resultant force for $\mu = 1$ is four times greater than for $\mu = 0$ which is consistent since the effective gap was divided by two and the force is inversely proportional to the square of the gap.

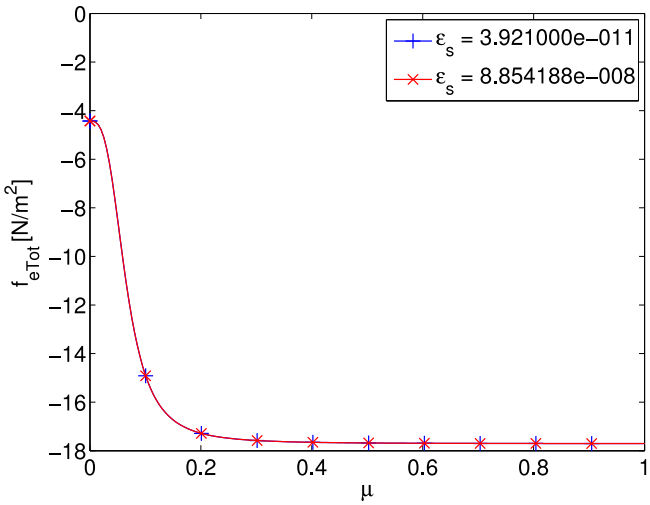


Figure 6.5: Total electrostatic force.

Besides the question of using or not two distinct permittivities, the last numerical experiment shows that the electrostatic force is a smooth and monotonous function of

the pseudo-density. This is a nice characteristic in the perspective of including the modeling procedure into an optimization problem. Moreover, in the present example, the use of a distinct dielectric constant for electrostatic forces computation has only local effects that compensate globally. However, because of the very simple geometry of the current problem this last conclusion cannot be generalized to all electrostatic models and to topology optimization problems. Consequently, this property has to be verified on more realistic models.

The main disadvantage of using different dielectric constants for voltage and force fields computation is that the electromechanical tangent stiffness matrix becomes asymmetric. This is in contradiction with the fact that the electromechanical monolithic formulation derives from the conservative Gibbs energy. Therefore, even if the method leads to more physically consistent electrostatic forces, it also reduces the physical relevance of the modeling method.

Moreover, since the tangent stiffness matrix is asymmetric, there is no more guarantee that the formulation of a stability eigenproblem, as described in Chapter 5, results in real eigenvalues. That's why this approach is not considered in this chapter and only generalization of the pull-in voltage design function is considered. Thus, the generalized optimization procedure can be subject to mode switching problems.

6.2.3 Material model choice

The set of material properties which have to depend on the pseudo-density is now determined. It includes the Young Modulus, the permittivity and the fictitious permittivity. However, the material model (i.e. the relation between each property and the pseudo-density) has still to be determined. In order to guide the definition of the material model, homogenization theory and practical considerations are first presented in this section. Then, the material model is established.

Hashin-Shtrikman bounds

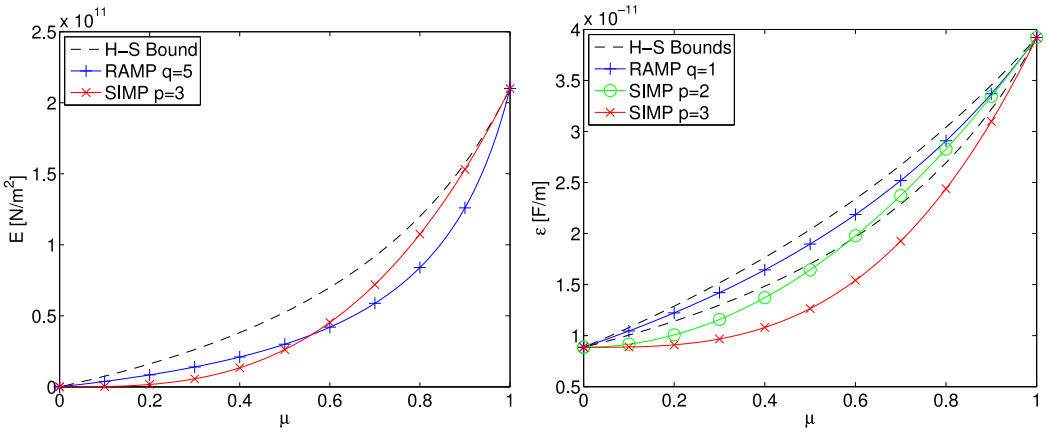
The homogenization approach to topology optimization is based on an interpolation of material properties between two materials, which are usually a design material and the void. This interpolation allows relaxing the optimization problem from discrete design variables to continuous design variables. One of the most famous and widespread interpolation functions is the SIMP power law whose main advantage is its simplicity of use. Moreover, even if the SIMP interpolation may seem artificial, Bendsøe and Sigmund [16] have shown that it was possible to define a microstructure whose homogenized mechanical behavior follows the SIMP model under some conditions on the Poisson ratio and parameter p value.

This paper by Bendsøe and Sigmund starts with a comparison of the SIMP law with Hashin-Shtrikman bounds. Hashin and Shtrikman worked on homogenization of effective properties of macroscopically homogeneous multiphase materials. Among others,

they have derived upper and lower bounds on effective mechanical properties of isotropic composite materials [46]. Hashin-Shtrikman bounds are functions of the volume ratio of the different phases that compose the composite and of the bulk mechanical properties of each phase.

By comparing these bounds to the SIMP model, Bendsøe and Sigmund show firstly that the lower bound is always satisfied. Indeed, because of the large ratio between the Young Moduli of the design material and of the soft material used to represent the void, the lower Hashin-Shtrikman bound is null over the whole range of μ. Secondly, they demonstrate that the upper bound is satisfied as soon as p is greater than a minimum value which depends on the Poisson ratio ν of the material (e.g. $p \geqslant 3$ for $\nu = 1/3$).

As an illustration, the Hashin-Shtrikman upper bound for plane elasticity is compared to SIMP and RAMP models in Figure 6.6(a). We can verify in this figure that the power function satisfies the upper bound for $p = 3$. Regarding RAMP law, according to conditions given by Stolpe and Svanberg [104], the upper bound is satisfied for values of q greater than 2.



(a) Young Modulus. (b) Permittivity.

Figure 6.6: Hashin-Shtrikman bounds for Young Modulus and permittivity.

Following an analog approach to the one used in Ref. [46], Hashin and Shtrikman in Ref. [45] have also established lower and upper bounds on the effective magnetic permeability of isotropic multi-phase materials. Because of the mathematical similarity between magnetic and electrostatic phenomena, these bounds are also valid for the dielectric constant. For a two-phase composite material, the bounds on the effective permittivity ε^* depend on the permittivity of the two phases ε_1 and ε_2 and on the volume fraction μ of phase 2. Then, the effective permittivity ε^* of an in-plane isotropic material has to satisfy (see [103]):

$$\varepsilon_1 + \frac{\mu}{\frac{1}{\varepsilon_2 - \varepsilon_1} + \frac{1-\mu}{2\varepsilon_1}} \leqslant \varepsilon^* \leqslant \varepsilon_2 + \frac{1-\mu}{\frac{1}{\varepsilon_1 - \varepsilon_2} + \frac{\mu}{2\varepsilon_2}} .$$

In the framework of our optimization problem, the first material is void with permittivity

$\varepsilon_1 = \varepsilon_0 = 8.85 \cdot 10^{-12}$ F/m while the second is quartz with permittivity $\varepsilon_2 = \varepsilon_s = 39.21 \cdot 10^{-12}$ F/m. Therefore the ratio between ε_2 and ε_1 is 4.4 which is far lower than the ratio between E_s and E_{min}. Consequently, conversely to the Young Modulus case, the lower bound for permittivity is non-zero and the admissible domain located between the two bounds is smaller as illustrated in Figure 6.6(b). The figure shows also that it is not possible to satisfy the bounds with SIMP model and an integer value of p since $p = 2$ violates the lower bound for low pseudo-density while $p = 1$ is greater than the upper bound. Conversely, RAMP function is able to satisfy both bounds when penalty q is included between 0.63 and 1.71. For these two extreme values the RAMP model coincides exactly with the Hashin-Shtrikman upper (resp. lower) bound.

From a global point of view, we can see that to satisfy homogenization theory a different interpolation function has to be used for each material property. For instance, a comparison of the admissible range of q for the RAMP model for both material properties shows that they do not overlap. Indeed, Hashin-Shtrikman upper bound on Young Modulus requires that $q \geqslant 2$, while the lower bound on permittivity imposes that $q \leqslant 1.71$. The conclusion is the same for the SIMP model. As a result, the penalty on Young Modulus should be higher than the penalty on permittivity.

Influence of the material model on local instability

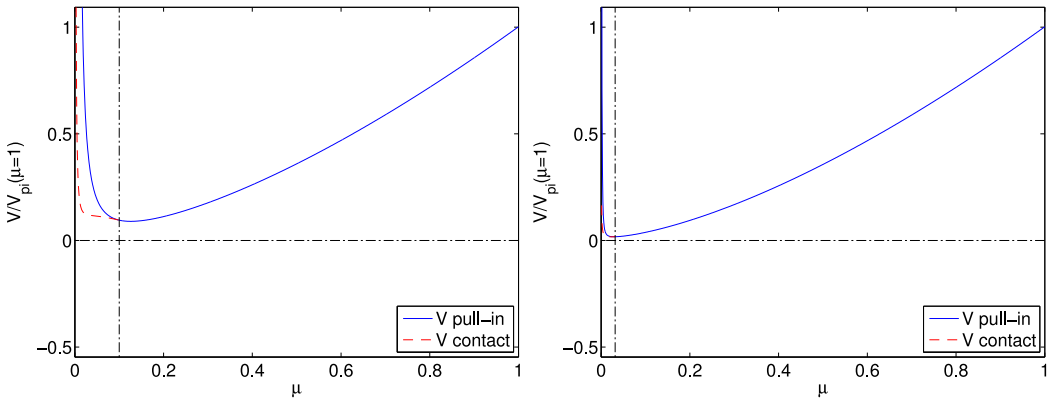
Nevertheless, it has been noted by Yoon and Sigmund [129] that in topology optimization context, the penalization on permittivity has to be at least as large as the one on Young Modulus. This statement comes from the practical observation of localized instability modes if this rule is not respected. Local instability modes have already been noticed in self-weight [27] or dynamic [83] problems. These modes may happen in topology optimization when low density areas of the optimization domain are submitted to a loading that is too large with respect to their stiffness. As a result, low density area exhibit large displacements even though they are not supposed to contribute to the response of the structure since they are not part of it. These local modes generally perturb the optimization process and lead to oscillations and break the convergence process.

The observations of Yoon and Sigmund can be explained using the one-dimensional system from Figure 6.2 in which mechanical behavior is added. The element located between nodes 0 and 1 is void and consequently has no stiffness. Conversely, the element included between nodes 1 and 2 is considered as elastic. Nodes 0 and 2 are fixed so that only node 1 is allowed to move. The Young Modulus of the elastic element is evaluated as a function of the density of the element μ using the SIMP model alike the two permittivities:

$$\begin{aligned} E(\mu) &= E_{min} + \mu^{p_E} (E_s - E_{min}), \\ \varepsilon(\mu) &= \varepsilon_0 + \mu^{p_\varepsilon} (\varepsilon_s - \varepsilon_0), \\ \tilde{\varepsilon}(\mu) &= \varepsilon_0 + \mu^{p_\varepsilon} (\tilde{\varepsilon}_{max} - \varepsilon_0). \end{aligned} \tag{6.6}$$

To investigate if local pull-in may happen in low density area, we propose to plot the pull-in voltage of the one-dimensional system against pseudo-density of the elastic element. Figure 6.7(a) presents the typical curve obtained with the material model used by Yoon and Sigmund (i.e. $p_E = p_\varepsilon = 3$). The pull-in voltage is normalized using its value for $\mu = 1$ as the reference value. Starting from the right of the figure for $\mu = 1$, a reduction of the pseudo-density results in a decrease of the pull-in voltage. Pull-in voltage possesses a minimum at 9% of the reference voltage. From this minimum, it tends to infinity if the density is decreased further.

However, for low densities, the behavior of the system is different because the permittivity of the elastic elements becomes very close to void permittivity. Whereas for pseudo-density equal to 1, the displacement of node 1 at pull-in is equal to $-s/3$. When pseudo-density is decreased pull-in displacement magnitude increases such that is equal to $-s$ for $\mu = 0.1$ (represented by the vertical dot dashed line). This means that for $\mu = 0.1$ pull-in instability occurs when node 1 comes in contact with node 0. For a lower value of the density, computed pull-in voltage is meaningless as it corresponds to displacement larger than $-s$ (i.e. larger than the gap). Therefore, on the left of the vertical dot dashed line the voltage corresponding to contact between node 1 and node 0 is plotted as a red dashed line. The contact voltage is always greater than the minimum value of pull-in voltage. Moreover, even if it remains smaller than pull-in voltage, it tends as well toward infinity for vanishing densities. In conclusion, a voltage equal to the tenth of the reference pull-in voltage leads to instability for $\mu = 0.1$. Moreover, the same voltage results in troublesome deformations for lower pseudo-densities.



(a) SIMP model with $p_E = p_\varepsilon = 3$.

(b) SIMP model with $p_E = 3$ and $p_\varepsilon = 2$.

Figure 6.7: Effect of the material model on stability of the one-dimensional system.

In the case where $p_E > p_\varepsilon$ the situation is even worse as shown in Figure 6.7(b). This figure is computed for $p_E = 3$ and $p_\varepsilon = 2$. Choosing p_ε equal to 2 gives a material model which is closer to the admissible range defined by Hashin-Shtrikman bounds for permittivity interpolation (see Figure 6.6(b)). The appearance of the pull-in voltage curve is similar to the previous case. However, its minimum value is smaller and equal to 1.7% of the reference pull-in voltage. Therefore, as noted by Yoon and Sigmund

[129], using lower penalization on permittivity than on Young Modulus increases the risk of local modes.

To limit the possibility of local modes, the opposite strategy could be used: to decrease penalization on Young Modulus in order to increase the material stiffness for low pseudo-densities. However, this solution only partially solves the problem as it is shown in Figure 6.8. Indeed, even if Young Modulus is linearly interpolated ($p_E = 1$), the pull-in voltage curve still possesses a minimum. Even if the value of this minimum can be increased further by increasing penalization on permittivity, its existence seems inherent to the problem. Therefore, the occurrence of local modes is always possible, no matter the material model used.

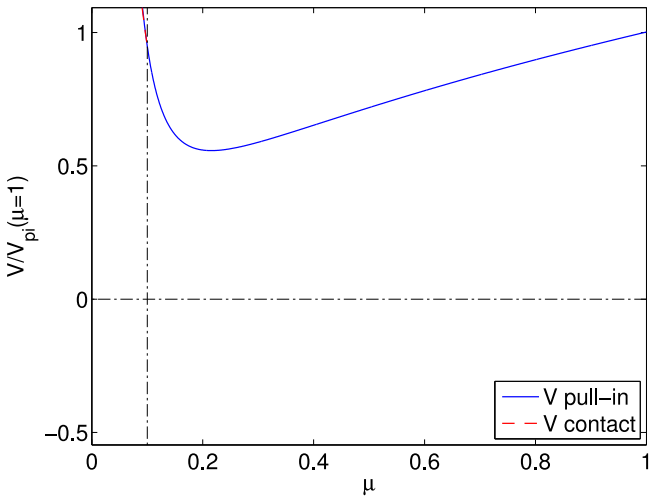


Figure 6.8: SIMP model with $p_E = 1$ and $p_\varepsilon = 3$.

However, the situation is also slightly different from what is observed in topology optimization considering self-weight or dynamic problems. In these two problems, the origin of local instabilities lies in the infinite value of the density to stiffness ratio for vanishing density. This results in very large deformation of non-structural parts of the domain for any load level. Conversely, in pull-in voltage optimization, local modes are only troublesome if the local pull-in voltage is close to or smaller than the actual pull-in voltage of the structure. Therefore, as shown by later numerical applications, if the Young Modulus and permittivity penalty are identical, local modes related to the material model do not appear in practice.

Summary

Homogenization theory and practical observation by Yoon and Sigmund lead to conflicting constraints for the selection of material model. Indeed, on one hand if Hashin-Shtrikman bounds have to be respected, Young Modulus should be more severely penalized than dielectric permittivity. But on the other hand when considering local

instability, Yoon and Sigmund recommend permittivity penalty to be at least as high as Young Modulus penalty. Therefore, as a trade-off between theoretical and application considerations, a penalty of 3 has been chosen for both material properties in subsequent applications.

6.3 Electrostatic forces computation

The last section provides guidelines for the choice of the material model in order to avoid the occurence of local instability modes. However, the present section shows that the material model is not the only source of local instabilities. At first, a topology optimization application illustrates that local modes can exist even though guidelines from last section are respected. Then, the origin of these local instabilities is investigated and it is shown that they are not only related to the material model but also to inaccuracy of computation of electrostatic forces.

6.3.1 Existence of local instability modes

The optimization procedure developed previously is tested on a simple problem presented in Figure 6.9 in order to show the appearance of local modes. In the figure, the design domain is represented by the gray area. The black rectangle in the middle of the optimization domain is the imposed electrode. The objective of the optimization problem is to design an optimal suspension for the imposed electrode using the supports provided on both sides of the optimization domain. The optimization domain is separated from the ground electrode by a non-designable void layer in order to prevent the creation of direct supports linking the ground electrode to the mobile electrode. Ground voltage (0 V) is defined on the bottom line of the electrostatic domain while input voltage is imposed on the mobile electrode and on both sides of the optimization domain. The available amount of material is limited to 25% of the design domain volume. Moreover, the problem is symmetrical and we also assume a symmetrical instability mode, thus only the left half of the domain is modeled.

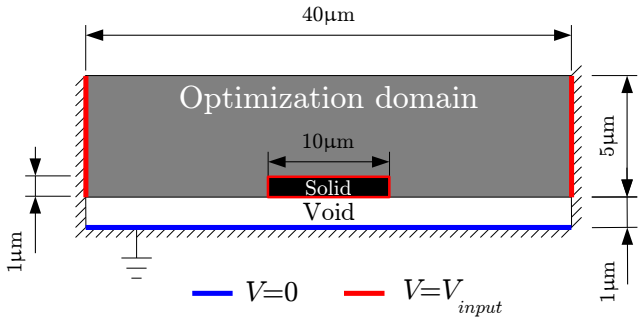


Figure 6.9: Geometry and boundary conditions for the first numerical application.

In this optimization problem, the optimization domain is in contact with the electrostatic domain. Therefore, the optimization domain is not insulated from the electrostatic effects unlike the numerical applications treated in Chapter 4. Thus, material distribution modifications have a direct influence on electrostatic forces distribution over the modeling domain.

	Void ($\mu = \mu_{min}$)	Quartz ($\mu = 1$)
Young Modulus [GPa]	$86.79 \cdot 10^{-6}$	86.79
Poison Ratio	0.17	0.17
Real permittivity [F/m]	$8.85 \cdot 10^{-12}$	$39.21 \cdot 10^{-12}$
Fict. permittivity [F/m]	$8.85 \cdot 10^{-12}$	$8.85 \cdot 10^{-9}$

Table 6.1: Material properties.

The first design material is quartz, which is placed where the density is unitary; the second material is void (for minimal densities). The properties of these two materials are listed in Table 6.1. Properties of intermediate density element are computed using the same material model as Yoon and Sigmund (see Eq. (6.6)). To comply with the conclusions from last section, the penalty parameters are the same for Young Modulus and permittivities and fixed to 3. The common penalty value results from a compromise between the Hashin-Shtrikman bounds on each physical property.

The optimization process is stopped after 200 iterations as it fails to converge. The material layout obtained at this point is presented in Figure 6.10. It is clear that the optimization process has not fully converged and gray areas remain over the optimization domain.

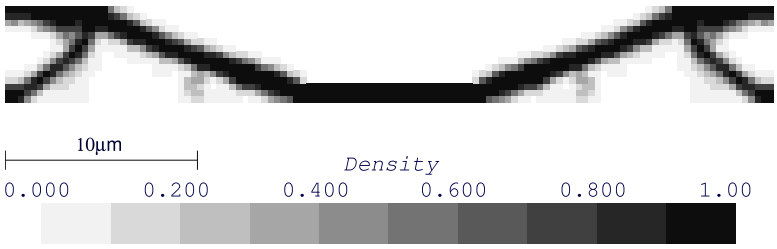
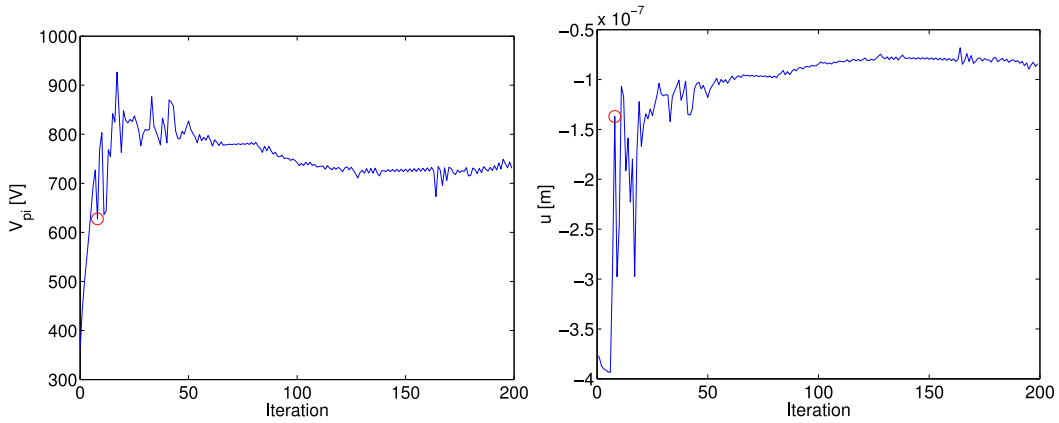


Figure 6.10: Final topology for the first numerical application.

Moreover, if we look at the history of pull-in voltage plotted in Figure 6.11(a), we can see that the behavior of the procedure becomes completely erratic after eight iterations. The explanation of this issue can be found by first considering the vertical displacement of the imposed electrode at the pull-in point found by the analysis procedure as plotted in Figure 6.11(b). During the very first iterations, this displacement is close to one third of the gap, which is a normal value considering that pull-in occurs on the imposed electrode. However, in the following iterations, the obtained pull-in configuration corresponds to a lower value of the electrode vertical displacement, which means that pull-in does not occur on the imposed electrode anymore.



(a) Pull-in voltage. (b) Pull-in displacement.

Figure 6.11: History of pull-in voltage and central node pull-in displacement across iterations.

The deformation of the mesh corresponding to pull-in state at iteration 8 (highlighted with a circle in Figures 6.11(a) and 6.11(b)) is presented in Figure 6.12. This iteration is the first one resulting in a drop of pull-in voltage and of the imposed electrode vertical displacement. The figure shows that the structure is globally almost undeformed. Conversely, a node located in the non-designable area of the modeling domain exhibits large displacements as illustrated by the close-up.

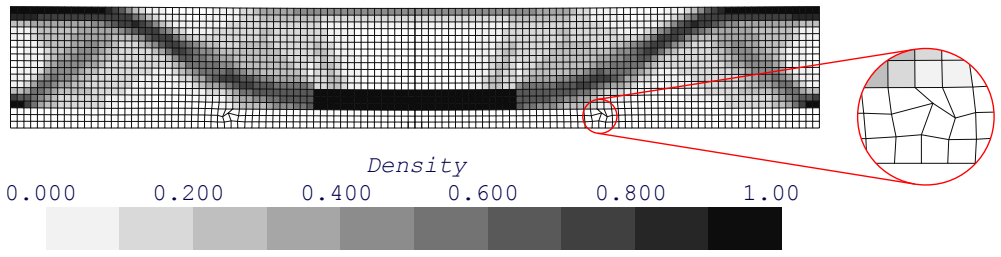


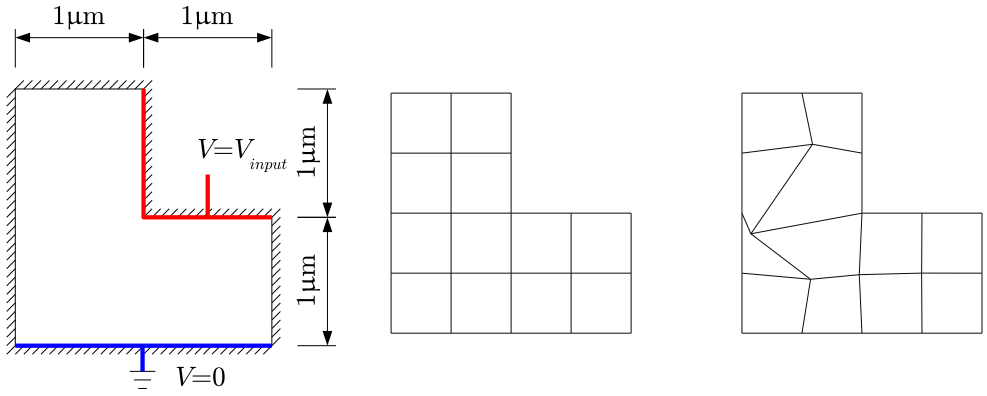
Figure 6.12: Illustration of the local pull-in mode.

Actually, it stems out that from the 8th iteration, the pull-in mode is not anymore implying the imposed mobile electrode but rather a few nodes located in a non-structural part of the modeling domain. This phenomenon looks similar to local modes that are commonly observed in topology optimization of eigenvalues. That is why we called it *local pull-in modes*. The existence of such local pull-in mode is a problem from the optimization point of view. Indeed, the computed pull-in configuration has no physical relevance and, of course, it cannot be used to compute the actual pull-in voltage sensitivities.

6.3.2 Origin of the local modes

The fact that a local mode appears in low stiffness areas could suggest a possible source of local modes as proposed in Section 6.2.3. Indeed, in these areas, small differences of permittivity between elements may generate electrostatic forces. Because of their low stiffness, the involved elements may not be able to withstand the electrostatic forces leading to a local pull-in. The solution would then require the adaptation of the material properties interpolation to adapt the stiffness to permittivity ratio.

However, it stems out that the local pull-in also occurs when permittivity is uniform over the domain and therefore that the interpolation is not the origin of the problem. To prove that conjecture, we have studied the behavior of an L-shaped electromechanical domain (see Figure 6.13(a)) over which the 'void' material used in the previous topology optimization example is applied uniformly (see Table 6.1). Therefore, no material model is used here since the void permittivity and a low Young Modulus are uniformly imposed over the modeling domain.



(a) Geometry and boundary conditions.

(b) Undeformed mesh.

(c) Deformed mesh.

Figure 6.13: L-shaped domain benchmark.

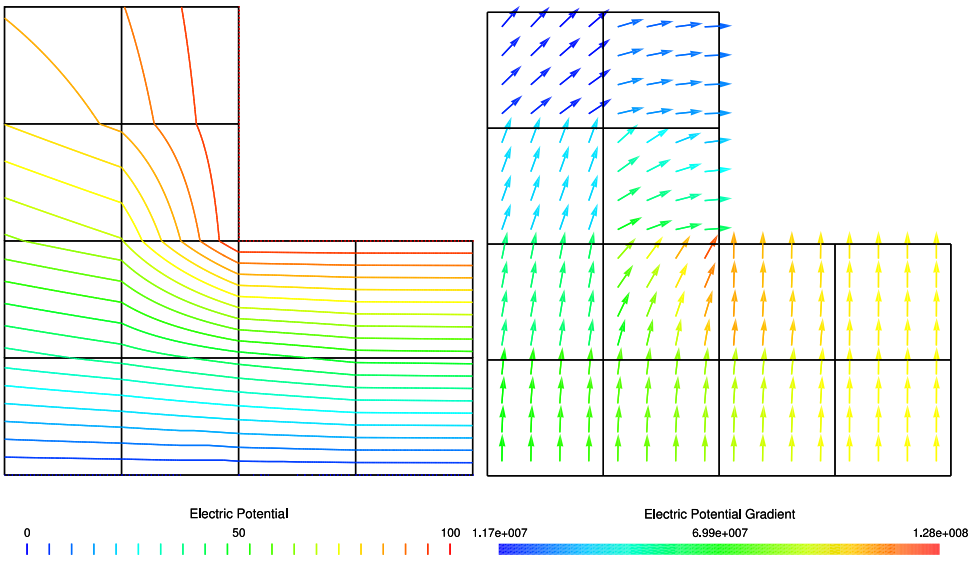
The choice of the L-shaped domain is driven by the observation that in last topology optimization local pull-in occurs close to a corner of the structure. This domain represents the void area that surrounds a structural corner. Considering boundary conditions, ground voltage is imposed on the lower edge of the modeling domain while input voltage is imposed on the two edges that form the inside corner. From a mechanical point of view, boundary nodes are clamped whereas the inner nodes are free to move. The domain is meshed with a coarse finite element mesh presented in Figure 6.13(b).

A pull-in search is then launched starting at $V_{input} = 0$ and increasing the voltage gradually while computing the equilibrium position of the inner nodes for each voltage. Normally the inner nodes should remain at the same place because material is uniform over the domain. Nevertheless, it is not the case and the pull-in search procedure actually finds a pull-in configuration where the elements are strongly distorted as shown

in Figure 6.13(c) for a voltage equal to 372 V. This behavior is very similar to local pull-in.

The displacement of the domain internal nodes betrays the existence of electrostatic forces on these nodes. However, as these nodes are surrounded by void elements, electrostatic forces should be zero on inner nodes. Therefore, to explain the origin of these electrostatic forces we have to examine the way they are evaluated. As described in [90], the electrostatic forces are computed at element level. A particular node is submitted to contributions from each element it belongs to. As already mentioned, the contributions from neighbor elements should normally cancel each other out for each inner node of the L-shaped domain. However, the electrostatic force is a function of the gradient of the electric potential. Therefore, the accuracy of the electrostatic force depends on the accuracy of the gradient of the finite element approximated field which is relatively poor with first order elements.

As an illustration, the voltage distribution resulting from a pure electrostatic model (i.e. inner nodes are fixed) of the L-shaped domain for $V_{input} = 100$ V is represented in Figure 6.14(a) and Figure 6.14(b) shows the gradient of electric potential. These figures have been obtained using *Gmsh* [42] improved post-processing that allows an accurate visualization of the finite element approximation of the voltage distribution [88]. As we can see in Figure 6.14(a), the first order finite element approximation ensures that the voltage distribution is continuous across finite element boundaries. Nevertheless, the isovoltage lines are non-smooth through element boundaries and result in discontinuities of the voltage gradient as it is clearly visible in Figure 6.14(b).



(a) Electric potential isovalue lines. (b) Electric potential gradient.

Figure 6.14: Electric potential isovalue lines on the L-shaped domain.

Mathematically speaking, the electric field is somehow analog to the strains in mechanics and there may be discontinuities between elements as the voltage is not represented

exactly by the finite element discretization. This is what is happening with the L-shaped domain; jumps of electric field at the boundary between elements lead to unbalanced electrostatic forces on the inner nodes of the domain.

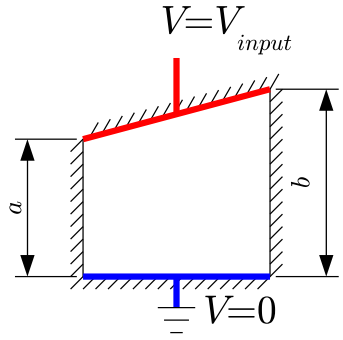
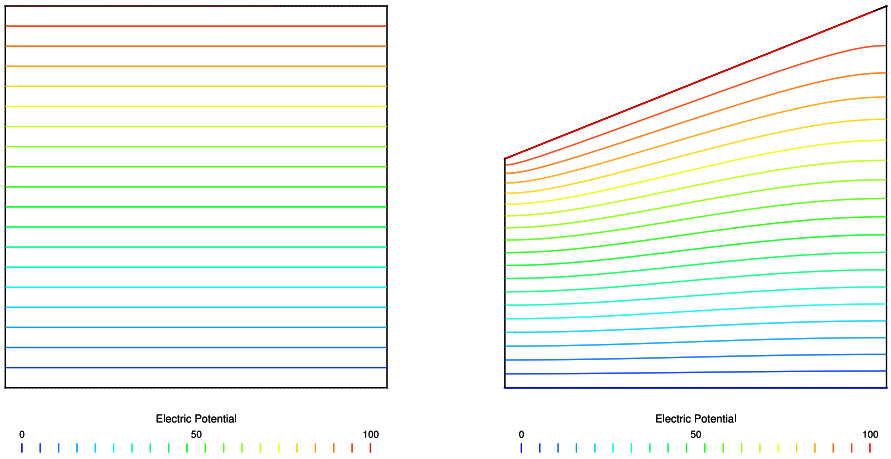


Figure 6.15: Trapezoid domain boundary conditions.

Conversely, in cases where the electrostatic field is exactly represented over the modeling domain, the phenomenon does not appear. For instance, if the modeling domain and boundary conditions presented in Figure 6.15 are considered, local pull-in is avoided as long as $a = b$ (i.e. the domain is rectangular). Indeed, under the condition of a rectangular domain, linear finite elements provide the exact solution as the voltage increases linearly from the ground electrode to the upper electrode and the electrostatic field is constant over the domain as shown in Figure 6.16(a).



(a) Quadrangular domain. (b) Trapezoidal domain.

Figure 6.16: Electric potential isovalue lines.

However, as soon as $a \neq b$, the domain becomes trapezoidal and a local pull-in can again be observed. Actually, when the domain is trapezoidal, the electric potential is not anymore linear as illustrated in Figure 6.16(b) that plots the voltage isovalues obtained with a very fine mesh (which is not represented for the sake of clarity). As one can see, the Neumann boundary conditions that are applied on both vertical sides of the

modeling domain impose the voltage isovalue lines to be locally perpendicular to these sides of the domain. As a result, the electric potential cannot be represented exactly by the finite element approximation and a local pull-in appears if boundary nodes are fixed while the inner ones are free. Therefore, we can conclude that the roots of the problem lie into the accuracy of the finite element approximation and cannot be solved by using an alternative material model.

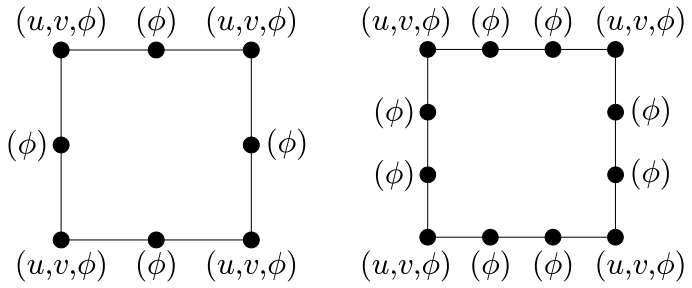
6.3.3 Solutions to prevent local modes

Higher order shape functions

Principle

Since local pull-in modes are related to a finite element accuracy problem, the first solution that is investigated consists in improving the accuracy of the electromechanical finite elements by using higher order shape functions and adding degrees of freedom. However, a simultaneous increase of the number of mechanical and electrostatic degrees of freedom would have an uncertain outcome. Indeed, one has to keep in mind that the electrostatic problem is solved on the deformed configuration. Adding mechanical degrees can result in a more complex deformed shape of the finite elements. This means that it would become harder to ensure the continuity of the electric field across element boundaries. Since our objective is to improve electric potential representation, the number of mechanical degrees of freedom should be kept constant while the number of electric potential degrees of freedom per element has to be increased.

Consequently, we choose to keep the mechanical interpolation degree as low as possible (bi-linear) and to increase solely the electrical interpolation degree. Following this procedure, two elements have been created in our finite element library. The first one has 8 voltage degrees of freedom (see Figure 6.17(a)) and uses quadratic shape functions; the second one owns 12 electric degrees of freedom (cubic shape functions) as shown in Figure 6.17(b). The shape functions used for electric potential can be found in Ref. [135,



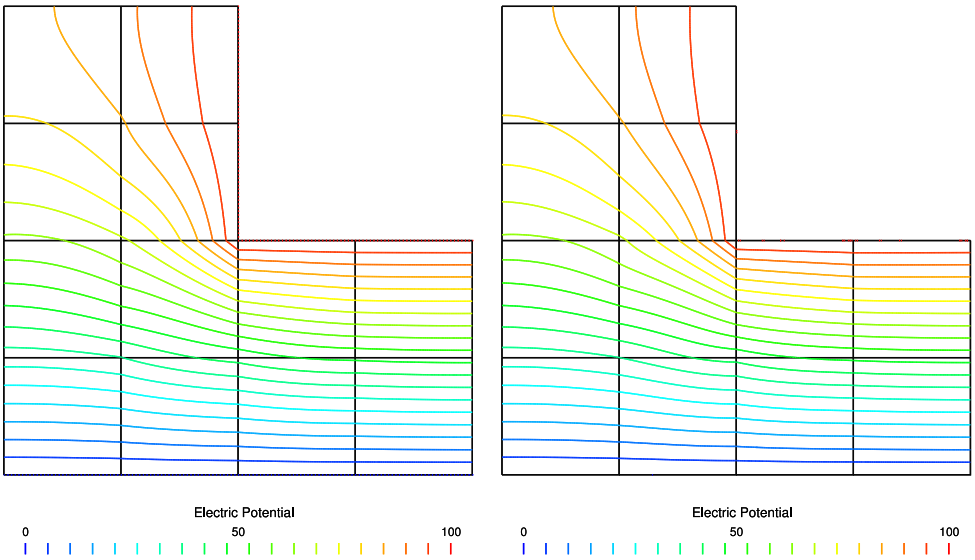
(a) Second order. (b) Third order.

Figure 6.17: Electromechanical finite elements with improved electrostatic representation.

Chap. 8]. In Figure 6.17, u and v denote the mechanical displacements and ϕ represents voltage degrees of freedom.

Application

To compare the two enriched finite elements with respect to the bilinear one, the L-shaped domain benchmark presented in Figure 6.13(a) has been solved using the proposed higher order elements. It comes out that the 8 nodes and 12 nodes elements improve the situation though they do not solve completely the problem. Indeed, local pull-in point can still be found with both higher order elements but the corresponding voltage is 493 V for the quadratic element and 518 V for the cubic element. Compared to the 372 V obtained with bilinear element, this corresponds respectively to an increase of 33% and 39%. The norm of the electrostatic forces acting on inner nodes follows the inverse trend. If this norm is computed for an input voltage $V = 100$ V, we have respectively: $3.04 \cdot 10^{-3}$ N/m, $1.14 \cdot 10^{-3}$ N/m and $1.01 \cdot 10^{-3}$ N/m for first, second and third order elements[1]. Hence, using higher order elements decreases the magnitude of the electrostatic forces acting on inner nodes and helps preventing local instability. The use of second order elements has a significant impact with respect to first order. However, with respect to second order elements, the additional improvement obtained with third order elements is smaller.



(a) Second order elements. (b) Third order elements.

Figure 6.18: Electric potential isovalue lines using higher order electric potential representation.

This result is related to the electric potential representation improvement that can be noted in Figures 6.18(a) and 6.18(b), which give the electric potential isovalues over the

[1]The norms are given per meter of the model thickness

modeling domain. By comparing these isovalue lines with the one obtained with bilinear element (see Figure 6.14(a)), we can see that we get here smoother lines. Nevertheless, if the improvement with respect to bilinear elements is obvious, the difference between second order element in Figure 6.18(a) and third order elements in Figure 6.18(b) is less clear as noticed with the local pull-in voltage.

The influence of higher order finite elements on the topology optimization problem presented in Figure 6.9 has also been tested. It results that the topology optimization process can be pushed slightly further ahead when using higher order elements but it cannot reach convergence so far. Pull-in voltage history during the first 60 iterations for both second and third order element is compared to the one provided by first order elements in Figure 6.19. The three curves are superposed up to the 8^{th} iteration. Then, while first order elements model is subject to local pull-in issue, second order and third order are able to achieve five more iterations before observing local pull-in modes on iteration 13. Moreover, the pull-in voltage associated with the first local pull-in mode at iteration 13 is 878 V with second order elements and 948 V with third order elements against 627 V for first order elements (at iteration 8). Again, the marginal improvement brought by third order elements is less than the one provided by passing from first order to second order elements.

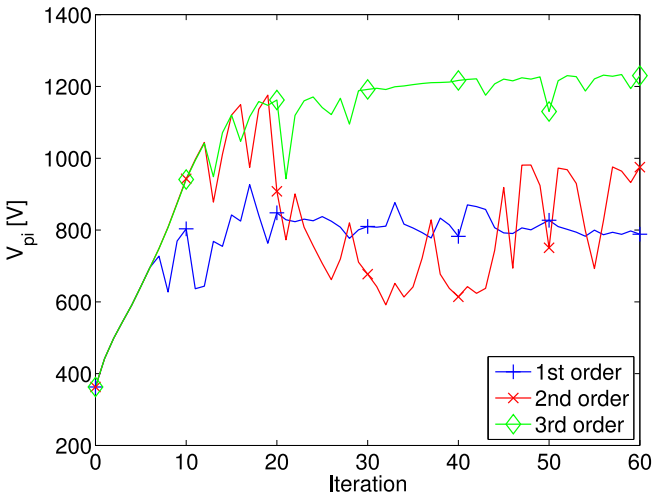


Figure 6.19: History of pull-in voltage across iterations.

Force filtering

Principle

As the element enrichment does not provide sufficient results; we consider another approach based on a filtering procedure that cancels electrostatic forces on nodes surrounded by void elements. The procedure consists in computing at each node a filtering factor, which is then applied to the electrostatic force. This filtering factor must be

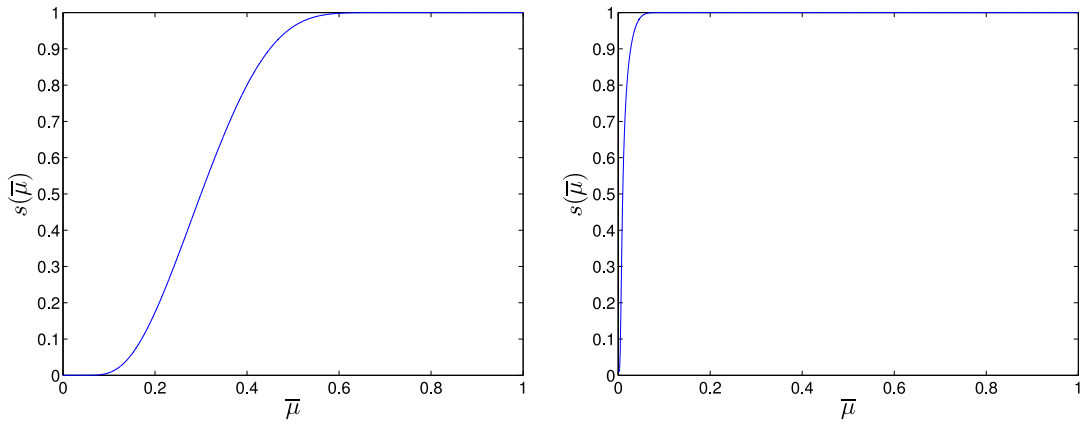
defined in such a way that the irrelevant electrostatic forces appearing on nodes surrounded by void are filtered out while the forces acting on nodes of the structural domain are not modified. Therefore, the filtering factor is computed as a function of the nodal density $\overline{\mu}$ defined as the average of the densities of the adjacent elements:

$$\overline{\mu} = \frac{1}{n} \sum_{i=1}^{n} \mu_i \,,$$

where n denotes the number of adjacent elements. To achieve its aim, the filtering factor should behave as a high pass filter. It has to take a low value when the nodal density is equal or close to the minimal density and equal to one when the nodal density is greater. Therefore, we suggest using the following function based on a hyperbolic tangent:

$$s\left(\overline{\mu}\right) = \frac{1}{2}\left(1 + \tanh\left(\frac{\overline{\mu} - \mu_{trans}}{\left(\overline{\mu} - \mu_{lb}\right)^{\eta_{min}} \left(\mu_{ub} - \overline{\mu}\right)^{\eta_{max}}}\right)\right) .$$

As an example the filtering function is plotted in Figure 6.20(a). The parameters appearing in this function allow some tuning. The location of the transition can be adjusted by modifying μ_{trans}. The sharpness of the transition can be adapted using η_{min} and η_{max}. Finally, parameters μ_{lb} and μ_{ub} correspond to the bound between which the density varies, usually one chooses μ_{ub} slightly greater than one and μ_{lb} slightly lower than the optimization problem minimal density to avoid division by zero.



(a) $\mu_{trans} = 0.3$, $\eta_{min} = 1$, $\eta_{max} = 2$, $\mu_{lb} = 9.999 \cdot 10^{-7}$ and $\mu_{ub} = 1.0001$.

(b) $\mu_{trans} = 0.01$, $\eta_{min} = 1$, $\eta_{max} = 20$, $\mu_{lb} = 9.999 \cdot 10^{-7}$ and $\mu_{ub} = 1.0001$.

Figure 6.20: Force filtering function with different parameters values.

On one hand, the force filtering restores the physical meaning of the model by removing irrelevant forces on nodes located in void areas. On the other hand, the filtering procedure is completely artificial and has to be carefully tuned to avoid reducing the physical foundations of the model. Therefore, it should not modify the electrostatic forces applied on nodes belonging to solid elements. However, on the other hand the

force filtering function cannot be reduced to a step function. Indeed, as we use gradient based optimization it is better to keep a sufficiently smooth filtering function to preserve differentiability. These requirements are conflicting and make the choice of the filtering function quite delicate.

Implementation

Moreover, in addition to the required tuning, the filtering procedure has another drawback and requires a few implementation modifications. Most of the modifications concern the electromechanical finite elements implementation. Firstly, the internal forces computation must be modified to reflect the effect of the filter. These forces are computed as the sum of the mechanical internal forces $\mathbf{f}_{m,int}$, the electrostatic internal charges $\mathbf{Q}_{es,int}$ and the electrostatic forces $\mathbf{f}_{es}$:

$$\mathbf{f}_{int} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_{m,int} \end{bmatrix} + \begin{bmatrix} \mathbf{Q}_{es,int} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_{es} \end{bmatrix}.$$

The force smoothing is introduced using a diagonal matrix $\mathbf{S}(\overline{\boldsymbol{\mu}})$ whose diagonal terms are equal to the value of the smoothing function of the associated node for each mechanical degree of freedom. To get the smoothed version of the internal forces, the electrostatic forces term is simply multiplied by $\mathbf{S}$ matrix:

$$\mathbf{f}_{int} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_{m,int} \end{bmatrix} + \begin{bmatrix} \mathbf{Q}_{es,int} \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{S}(\overline{\boldsymbol{\mu}})\,\mathbf{f}_{es} \end{bmatrix}.$$

As a consequence, the pull-in voltage sensitivity evaluation procedure has to be adapted since it implies the internal forces sensitivities as obtained in Eq. (4.2). As matrix $\mathbf{S}$ depends on the densities, an additional term appears in the internal forces sensitivities. Nevertheless this term is relatively simple to evaluate as it only involves the derivative of the smoothing function and the electrostatic forces.

Moreover, to keep the modeling procedure consistent, the modification of the internal forces must result in a modification of the tangent stiffness matrix. Indeed, the tangent stiffness matrix corresponds to the derivative of the internal forces with respect to the generalized displacements. Therefore, each block of the tangent matrix corresponds to the derivative of a particular contribution to the internal forces. If we use the same notation as in Eq. (2.6):

$$\mathbf{K}_t = \begin{bmatrix} \mathbf{K}_{\phi\phi} & \mathbf{K}_{\phi u} \\ \mathbf{K}_{u\phi} & \mathbf{K}_{uu} \end{bmatrix} = \frac{\partial \mathbf{f}_{int}}{\partial \mathbf{q}} = \begin{bmatrix} \frac{\partial \mathbf{Q}_{es}}{\partial \boldsymbol{\phi}} & \frac{\partial \mathbf{Q}_{es}}{\partial \mathbf{u}} \\ \mathbf{S}\frac{\partial \mathbf{f}_{es}}{\partial \boldsymbol{\phi}} & \mathbf{S}\frac{\partial \mathbf{f}_{es}}{\partial \mathbf{u}} + \frac{\partial \mathbf{f}_{m,int}}{\partial \mathbf{u}} \end{bmatrix}.$$

While the original form of the tangent stiffness matrix is symmetric as we have from Eq. (2.6) (in the case where no modification of the permittivity is considered):

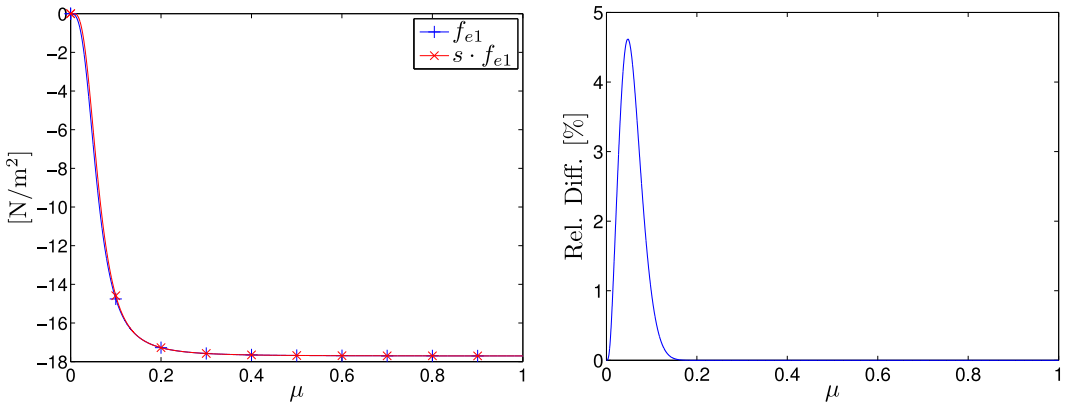
$$\frac{\partial \mathbf{f}_{es}}{\partial \boldsymbol{\phi}} = -\frac{\partial}{\partial \boldsymbol{\phi}}\frac{\partial W_e}{\partial \mathbf{u}} = -\frac{\partial}{\partial \mathbf{u}}\frac{\partial W_e}{\partial \boldsymbol{\phi}} = \frac{\partial \mathbf{Q}_{es}}{\partial \mathbf{u}}.$$

The smoothing matrix appears now in $\mathbf{K}_{u\phi}$, which means that the force smoothing procedure introduces another source of asymmetry into the tangent stiffness matrix (the first one being the asymmetry caused by the fictitious permittivity).

Application

At first, the parameters of the smoothing function need to be chosen. This choice is performed on the basis of the 1D example described in Section 6.2.2. For this example, the electrostatic force has been computed analytically and thus is free from numerical artifacts. This 1D example has shown that intermediate density elements may generate significant electrostatic force even for low density (see Figure 6.4(b)). However, as our topology optimization procedure uses density filtering, the boundary of the structure contains a lot of low and intermediate density elements. Therefore to avoid removal of forces acting on low density elements, the range for which $s(\overline{\mu}) = 0$ has to be very narrow as the one obtained in Figure 6.20(b) by setting μ_{trans} to 0.01 and η_{max} to 20 while η_{min} is left to 1 to keep the filtering function as smooth as possible. Notice that the plot abscissa is the average density so that the function is smoother if plotted with respect to one element density.

To determine the effect of the smoothing function represented in Figure 6.20(b), the smoothing procedure has been applied to the 1D electrostatic example considered in Section 6.2.2. Figure 6.21(a) plots the original (analytical) value of the electrostatic force at node 1 and its product with the smoothing function with respect to the density of the varying density element. Because of the very low value selected for μ_{trans}, the effect of the force smoothing is barely visible in this figure. The effect of the smoothing



(a) Comparison of original and smoothed electrostatic force against density.

(b) Relative difference: $\frac{f_{e1} - s \cdot f_{e1}}{f_{e1}(\mu=1)}$.

Figure 6.21: Effect of the smoothing function on the electrostatic force.

procedure is more visible by plotting the ratio of the difference between the original and the smoothed force with the maximum value of the force ($f_{e1}(\mu = 1)$) as proposed in Figure 6.21(b). This figure shows that the force smoothing procedure impacts a larger

range of density than what can be expected from the plot of the smoothing function (Figure 6.20(b)). The reason of this difference lies into the choice of the abscissa as the smoothing function is plotted against the (nodal) average density while the forces are plotted against the element density. As we can see in both figures, the selected smoothing function has a very mild impact on the electrostatic forces. Therefore it has been selected in order to study the efficiency and the impact of force smoothing procedure on the numerical application presented in Figure 6.9.

Thanks to the addition of the smoothing procedure, it is possible to solve the optimization problem from Figure 6.9 up to convergence. The final structure is presented in Figure 6.22 and its pull-in voltage is equal to 1262 V after 143 iterations. The filtering procedure allows reaching higher pull-in voltage than the previous enrichment method without appearance of local pull-in effect. In addition to the topology, the voltage isovalue lines are also plotted in Figure 6.22. As the optimization domain is now multiphysic, the isovalue lines are allowed to penetrate into the optimization domain. Moreover, we can check that the value of the fictitious permittivity $\tilde{\varepsilon}$ is high enough as the voltage is uniform over the structure.

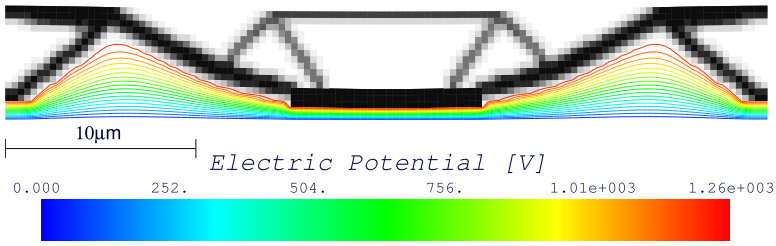


Figure 6.22: Optimization problem result using electrostatic force filter.

The history of pull-in voltage during the first optimization iterations is presented in Figure 6.23 and compared to the history obtained with first order and third order elements. At first, we can note that introducing force smoothing into the optimization procedure stabilizes greatly its convergence. Secondly, we can see that the smoothed solution coincides with first order and third order solution until each of the former methods fail because of local pull-in. This is a good indicator that the parameters of the force smoothing procedure are well adapted so that the procedure does not modify the physical behavior of the model.

In order to verify the validity of the optimization results obtained here, an additional analysis has been performed on the final design in order to test the influence of the filtering procedure. As it is not possible to reach structural pull-in if the force smoothing procedure is disabled, the function parameters have been modified so that it becomes very sharp and only zeroes for very small nodal densities. In the context of a post-optimization analysis, it is not necessary to keep the $s\left(\overline{\mu}\right)$ function as smooth as possible and its sharpness is not a problem anymore. Therefore, the parameters are fixed to $\mu_{trans} = 10^{-4}$, $\eta_{max} = 1000$ and $\eta_{min} = 1$, which result in a relative difference between original and smoothed forces of 10^{-4} % if computed as in Figure 6.21(b). The pull-

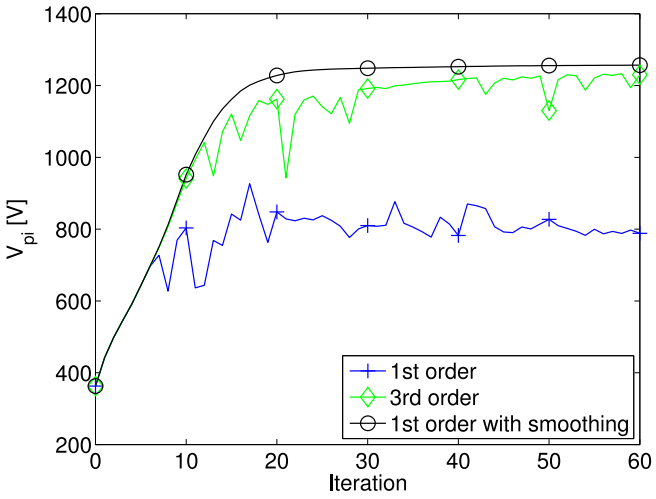


Figure 6.23: Pull-in voltage history.

in point search results in a pull-in voltage of 1261 V which is very close to the value obtained at the end of the optimization procedure. Therefore, we can conclude that the force smoothing function used for optimization does not involve significant modifications of the model behavior while it avoids the appearance of local pull-in modes.

6.4 Numerical application

In this section a second application is proposed in order to show the efficiency of the developed method. Design materials are the same as in the previous optimization problem. The design problem geometry is presented in Figure 6.24. The domain presents a large non-designable area located between the top supports and the imposed electrode. Notice also that one part of the area located below the imposed electrode is also part

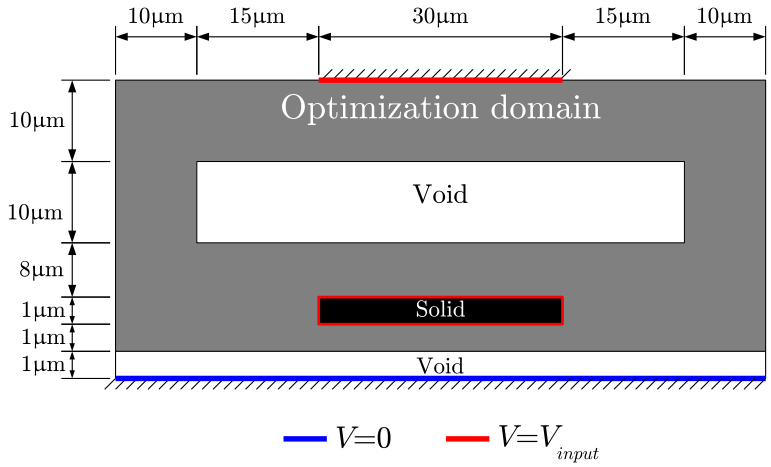
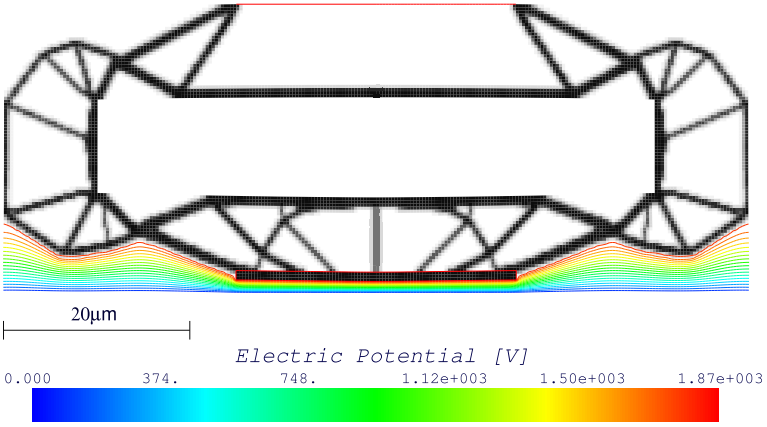


Figure 6.24: Geometry and boundary conditions of the second optimization problem.

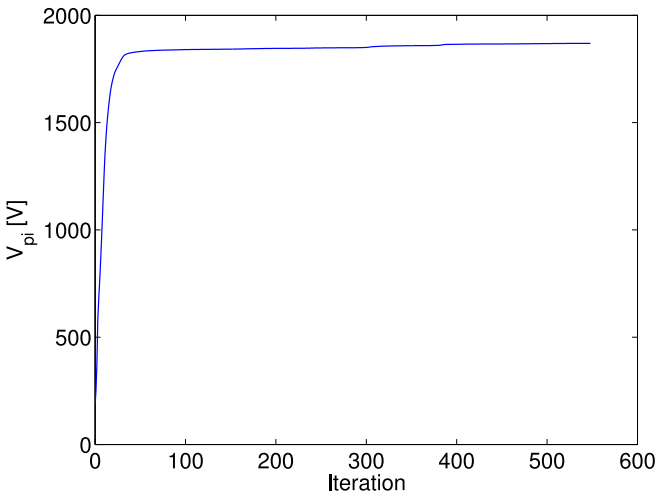
of the design domain. The maximum amount of material is fixed to 20% of the optimization domain volume. As the problem is symmetrical again, only the left half of the domain is modeled. The element size is set to one third of micron, which leads to 7965 design variables.

6.4.1 Pull-in voltage optimization

The filtering method is also used here with the same parameters as it provides good results for the first application. The resulting structure is presented in Figure 6.25(a). The design procedure leads to a relatively complex suspension system that surrounds the non-designable void area. Moreover, logically, the designable area located directly below the imposed electrode is void in order to maximize the distance with respect to the ground electrode. As presented in Figure 6.25(b), the optimization process is



(a) Final design.



(b) Pull-in voltage history.

Figure 6.25: Pull-in optimization results for the second application.

relatively slow to reach convergence as 547 iterations are needed to satisfy the stopping criterion. The final pull-in voltage of the design is equal to 1869 V.

6.4.2 Comparison with linear compliance optimization

In order to highlight the interest of developing pull-in optimization procedure, the present problem has been carried out considering linear compliance objective function similarly to what is proposed in Section 4.3.3. The load case acting on the structure is determined once for all optimization iterations by computing the electrostatic forces that lead to pull-in if the optimization domain is void. This force distribution is presented together with compliance optimization result in Figure 6.26.

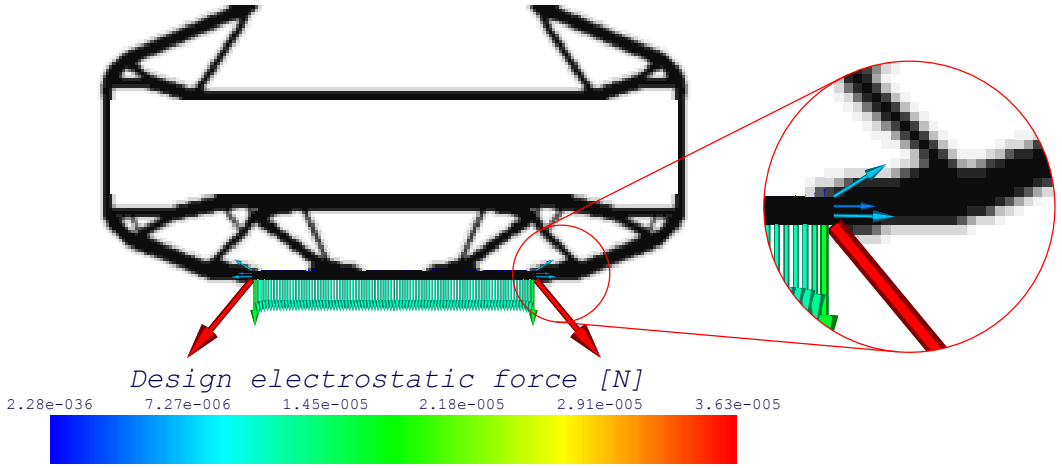


Figure 6.26: Linear compliance optimization result and design load case.

Compliance optimization leads to a different design from pull-in voltage optimization as lateral stiffening structures are missing. Moreover, conversely to pull-in optimization, compliance optimization adds horizontal extensions on both sides of the imposed electrode as indicated by the red circle and the close-up in Figure 6.26. Since they are very close form ground electrode, these extensions are penalizing regarding the final pull-in voltage of the design. Indeed, as illustrated in Figure 6.27, the pull-in electrostatic force distribution is very different from the original one (used for compliance design). The pull-in voltage of compliance design is equal to 1582 V which means 18% lower than the value reached by pull-in voltage optimization. Conversely, the compliance of the pull-in optimized design in Figure 6.25(a) is 1% greater than the compliance of present structure. This confirms that both optimization problems lead to specialized designs.

The physical justification of lateral structures in pull-in optimization design is unclear. Indeed, from one point of view, these structures increase the bending stiffness of this part. But from another, they add electrostatic loading to the device even though the magnitude of these forces is lower than the one created by the imposed electrode. Therefore one may argue that a structure similar to the one obtained by compliance design (see

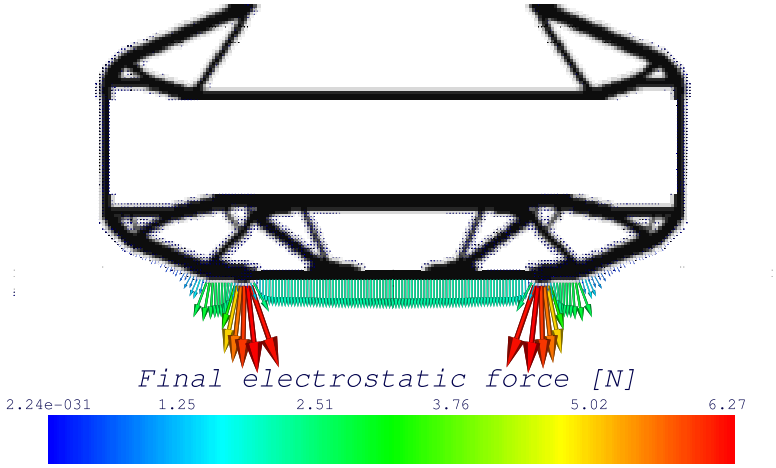


Figure 6.27: Actual electrostatic force distribution on the linear compliance result.

Figure 6.26) could be more efficient in terms of pull-in voltage than the one presented in Figure 6.28. To verify this, pull-in optimization procedure has been launched using the compliance design as initial design. In order to make sure that the optimization process reaches convergence, the procedure is pursued during 5000 iterations.

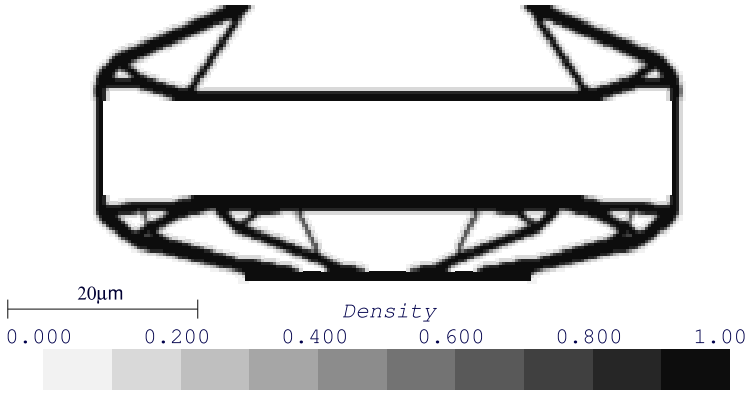


Figure 6.28: Optimization result for the second application.

After this large number of iterations the design and the objective function do not evolve anymore. The obtained design is presented in Figure 6.28, this structure is different from previous pull-in voltage design. The horizontal extensions of the mobile electrode are removed but the lateral structures do not appear. Its pull-in voltage is equal to 1850 V that is to say 1% less than the previous pull-in voltage design. The present structure corresponds to a local optimum of the optimization problem that is slightly less efficient than the original one. Starting from the compliance optimized design pull-in optimization procedure get stuck into this local optimum and is not able to reach the original design presented in Figure 6.25(a).

6.5 Conclusion

The generalization of the optimization problem proposed in this chapter allows increasing the design freedom of the optimization procedure. The shape and the topology of the structure have now a direct impact on the electrostatic forces distribution. To achieve this improvement, the permittivity of each element is computed as a function of the pseudo-density. In order to keep the model as close to physics as possible, we attempted to select the permittivity interpolation function such that it complies with Hashin-Shtrikman bounds.

However, as shown with a simple 1D example, it is not possible to satisfy Hashin-Shtrikman bounds on permittivity since it promotes local instability. Indeed, Young Modulus and permittivity interpolations have to be carefully chosen so that low stiffness element are not submitted to high electrostatic loading. Otherwise, local instability modes may appear similarly to what happens in topology optimization problems involving self-weight or dynamic effects. As observed by Yoon and Sigmund, the apparition of local pull-in can be limited by selecting the permittivity penalization greater or equal to Young Modulus penalty.

Unfortunately, the first numerical application shows that the material model is not the only source of local instability. Further investigation shows that local pull-in can appear even if the material properties are uniform over the modeling domain because of the approximations underlying finite element method.

These local modes are a serious issue for topology optimization of pull-in voltage as they prevent from computing the actual pull-in conditions of the structure and the associated sensitivities. Different solutions to overcome the problem have been investigated. Since the local mode existence is linked to a finite element precision problem, we have first attempted to remove them by improving the electrical voltage representation of the elements. Electrostatically quadratic and cubic elements have been developed. Nevertheless, if these elements delay the appearance of local modes, they do not suppress them and local modes appear anyway.

Therefore, a second solution has been developed. This one is based on a filtering of the electrostatic forces that removes these forces on nodes surrounded by void elements. This solution seems more drastic but somehow it restores the physical meaning of the model by removing the electrostatic forces resulting from modeling inaccuracies. However, it has to be used carefully and the filtering function has to be properly tuned to prevent from perverting the physics of the electrostatic forces.

Nevertheless, force filtering method allows solving the optimization problem and provides consistent results as shown by two numerical applications. A comparison with linear compliance optimization shows that conversely to the simplified optimization problem the present generalization leads to results which are very different from the one that can be obtained using the linear optimization method.

Hitherto, a simple optimization problem which consists in maximizing pull-in voltage

with a limited amount of material is considered. This optimization problem is convenient for developing and testing the optimization procedure but its practical interest is limited. Therefore, on basis of the gained experience, next chapter proposes to study a more realistic optimization problem where pull-in voltage function appears in a constraint.

Chapter 7

Electrostatic actuator optimization

7.1 Introduction

The design problem of electrostatic actuator has been previously studied by Raulli *et al.* [86] and Yoon *et al.* [129] as presented in Section 3.5.2. However, this optimization problem may benefit from the developments proposed along the previous chapters which allow including pull-in voltage into an optimization problem. Indeed, under some circumstances, the original electrostatic actuator design problem as formulated by Yoon and Sigmund fails to reach convergence because of pull-in effect. On the basis of the developments proposed in this thesis, the optimization problem can be improved in order to include a constraint on the pull-in voltage which prevents instability from appearing during optimization.

We start this chapter by presenting our own implementation of the electrostatic actuator optimization problem proposed by Yoon and Sigmund. Then, on the basis of a numerical application, we show that Yoon and Sigmund's approach may fail because of pull-in instability. Using a simplified optimization problem, we explain why the appearance of pull-in effect is inherent to the optimization problem.

Therefore, the proposed solution to this issue consists in introducing an additional design constraint in the optimization problem. This new constraint prevents pull-in voltage from decreasing below the driving voltage and preserves the microsystem from instability. This modification of the optimization problem takes advantage of the achievements from the previous chapters. Also, it allows demonstrating the usefulness of pull-in voltage function in the context of a practical optimization problem. The efficiency of the proposed optimization problem is demonstrated thanks to a numerical application and the need for considering large displacement modeling hypothesis is investigated.

7.2 Electrostatic actuator design procedure

7.2.1 Optimization problem

Generally speaking, actuators are devices able to convert a given form of energy (e.g. electric, thermal or chemical) into mechanical energy. The produced mechanical energy is transmitted to the workpiece(s) through the output port(s) of the actuator. The classical topology optimization problem for actuator design is inspired from compliant mechanisms topology optimization (see Ref. [98]). It consists in maximizing the energy provided at the output port for a given input. The workpiece is modeled by a spring of constant stiffness connected to the output port. Equivalently to maximizing the output mechanical energy, we can choose to maximize either the output displacement or the output force.

The electrostatic actuator optimization problem is sketched in Figure 7.1. As in Chapter 6, the optimization domain is electromechanical. The actuator is designed for a fixed input voltage V_{input}. This voltage and the ground voltage are provided through imposed electrodes. Moreover, the workpiece is modelled as a linear spring of stiffness k attached to the requested output port.

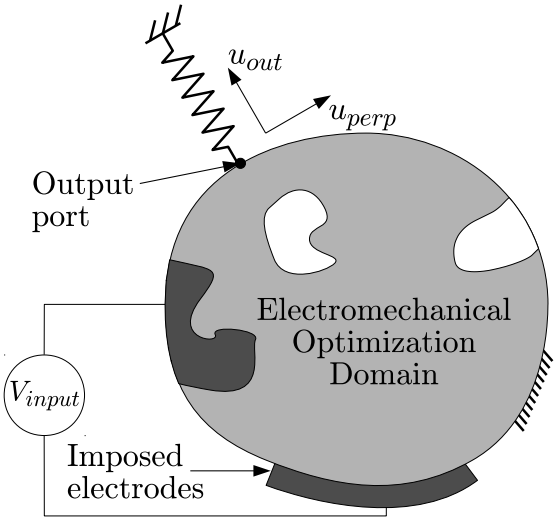


Figure 7.1: Schematic view of the electrostatic actuator optimization problem.

The optimization problem is mathematically formulated in Eq. (7.1). The objective of the optimization problem is to maximize the output port displacement u_{out} along the spring direction. In addition as suggested by Sigmund in [99], a constraint is added in order to limit u_{perp}, the output port displacement component perpendicular to the spring. This constraint limits the magnitude of the ratio u_{perp}/u_{out} to a small value e such that the final design operates along the spring direction. Finally, an optional volume constraint can be added to the optimization problem. Strictly speaking, the volume constraint is unnecessary for actuator topology optimization but it prevents

from getting designs with material islands and gray areas (see Ref. [99]).

$$\max_{\boldsymbol{\mu}} u_{out}\left(\boldsymbol{\mu}, V_{input}\right) \ , \quad s.t. \begin{cases} \left(\frac{u_{perp}}{u_{out}}\right)^2 \leqslant e^2, \\ v\left(\boldsymbol{\mu}\right) \leqslant \overline{v} \qquad \text{(Optional)}, \\ \mu_{\min} \leqslant \mu_i \leqslant 1 \qquad \forall i. \end{cases} \tag{7.1}$$

As already mentioned, actuators design and compliant mechanisms design are related optimization problems. However, previous studies on topology optimization of compliant mechanisms by Bruns and Tortorelli [25] and Pedersen *et al.* [82] show that the modeling of these devices using small displacement hypothesis is generally invalid. They conclude that large-displacement (i.e. geometric nonlinearity) modeling should be used in order to get physically consistent designs. Moreover, the use of large-displacement analysis even leads to more efficient designs at the price of numerical difficulties related to negative defined stiffness matrix in void areas. Therefore, in this chapter, small displacement hypothesis is considered at first and the necessity to go to large-displacement analysis is investigated later on the basis of the first results.

7.2.2 Sensitivity analysis

The sensitivity analysis can be developed by using either the adjoint approach [99] or the direct approach, which both lead to the same final expression. We choose here to describe the direct approach. A generic notation u_* for displacement function is used to represent either u_{out} or u_{perp}. The displacement function can be defined as:

$$u_* = \mathbf{L}_*^T \mathbf{q},$$

where $\mathbf{L}_*$ is specific to a particular function u_*. It is a vector filled with zeros and a '1' to select appropriate component of the generalized displacements vector $\mathbf{q}$. We start by deriving the expression of the displacement function:

$$\frac{\partial u_*}{\partial \mu_i} = \frac{\partial \mathbf{L}_*^T \mathbf{q}}{\partial \mu_i} = \mathbf{L}_*^T \frac{\partial \mathbf{q}}{\partial \mu_i}.$$

Where the derivative of the generalized displacements appears. To evaluate this derivative, we make the assumption that the equilibrium is satisfied (i.e. $\mathbf{r} = \mathbf{0}$) and derive the equilibrium equation:

$$\frac{d\mathbf{r}}{d\mu_i} = \mathbf{0} = \frac{\partial \mathbf{r}}{\partial \mu_i} + \underbrace{\frac{\partial \mathbf{r}}{\partial \mathbf{q}}}_{=\mathbf{K}_t} \frac{\partial \mathbf{q}}{\partial \mu_i} + \underbrace{\frac{\partial \mathbf{r}}{\partial V}}_{=\mathbf{0}} \frac{\partial V}{\partial \mu_i}.$$

Since the considered problem is free from design dependent external forces, the first term is equal to the sensitivity of the internal forces, by definition of the residual forces vector in Eq. (2.11). In the second term, we can identify the tangent stiffness matrix.

At last, the third term is null in the present case because the input voltage is fixed for evaluating the output displacement. As a result, we can write:

$$\frac{\partial \mathbf{q}}{\partial \mu_i} = -\mathbf{K}_t^{-1} \frac{\partial \mathbf{r}}{\partial \mu_i} = -\mathbf{K}_t^{-1} \frac{\partial \mathbf{f}_{int}}{\partial \mu_i}.$$

Finally, by substituting the last expression into the displacement derivative, we obtain the expression of the displacement function sensitivities:

$$\frac{\partial u_*}{\partial \mu_i} = -\mathbf{L}_*^T \mathbf{K}_t^{-1} \frac{\partial \mathbf{f}_{int}}{\partial \mu_i}. \tag{7.2}$$

From a computational point of view, when considering a topology optimization problem, it is more efficient to compute once $\mathbf{L}_*^T \mathbf{K}_t^{-1}$ for each function u_* than computing $\mathbf{K}_t^{-1} \frac{\partial \mathbf{f}_{int}}{\partial \mu_i}$ for every design variable. Indeed, in the topology optimization context the number of functions u_* is usually lower than the number of design variables. Consequently, we retrieve a sensitivity expression similar to the one obtained in [99] using the adjoint approach.

7.2.3 Optimization procedure implementation

The optimization procedure flowchart is presented in Figure 7.2. The general structure of the flowchart is identical to the pull-in maximization one presented in Chapter 4 (see Figure 4.3). However, the analysis phase is different. In the present optimization problem, there is no need to compute pull-in configuration but only the generalized displacements resulting from the input voltage $\mathbf{q}(V_{input})$. Therefore, the normal flow analysis is replaced by a Newton-Raphson algorithm (see [136, Chap. 2]) which is simpler and better suited to compute the response for a fixed voltage.

7.3 Numerical application

7.3.1 Optimization problem

The electrostatic actuator design problem is presented in Figure 7.3. The modeling domain is divided into two parts. The first one, on the right is the optimization domain and the second one on the left is a void area that can be considered as the outside world. Ground voltage is provided along the bottom side of the optimization domain while input voltage is imposed on a part of its top side. The problem is symmetrical and only one half of the problem is modelled[1]. Only the left side of the optimization domain is completely free to move. The objective of the optimization problem is to maximize the displacement of the output port indicated in Figure 7.3 towards the inside of the optimization domain.

[1]To be rigorous, this means that asymmetric instability modes cannot be captured by the model and that we assume a symmetrical instability mode

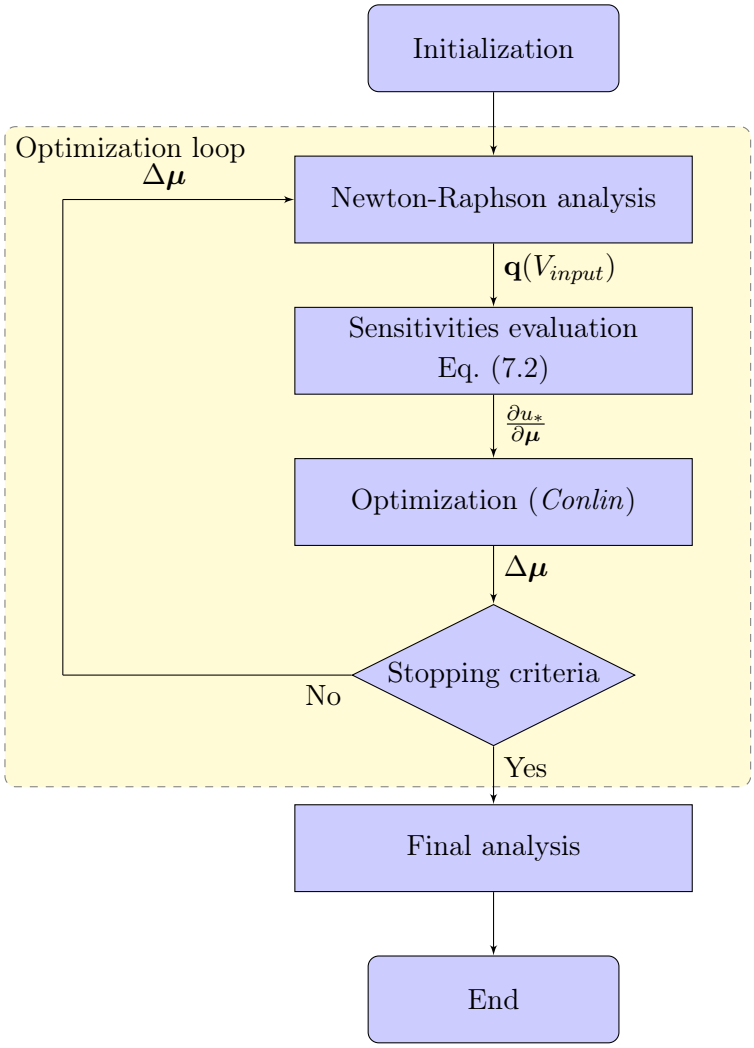


Figure 7.2: Output displacement optimization procedure flowchart.

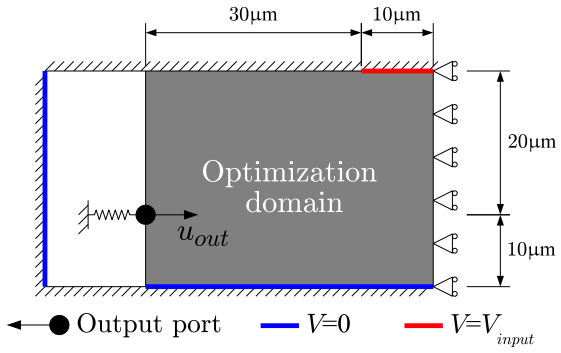


Figure 7.3: Actuator design problem sketch.

The properties of the design material are listed in Table 7.1. Except the quartz fictitious permittivity, these properties are identical to the ones used in applications from Chapter 6. The fictitious permittivity has to be modified because of the hinges that appear in the electrostatic actuator design. The optimizer makes these hinges as thin as possible such that they become non-conducting. The creation of such hinges is common when considering topology optimization of actuators (see for instance [99]). But, since hinges are part of the structure, they should be conducting and a higher fictitious permittivity is used to ensure consistent behavior. Similarly to the numerical applications from Chapter 6, the SIMP model with $p = 3$ is used to compute the properties of intermediate density elements (Eq. (6.6)). Additionally, force filtering is also used with the same parameters as the ones used to solve the applications in Chapter 6 (see Figure 6.20(b)).

	Void ($\mu = \mu_{min}$)	Quartz ($\mu = 1$)
Young Modulus [GPa]	$86.79 \cdot 10^{-6}$	86.79
Poison Ratio	0.17	0.17
Real permittivity [F/m]	$8.85 \cdot 10^{-12}$	$39.21 \cdot 10^{-12}$
Fict. permittivity [F/m]	$8.85 \cdot 10^{-12}$	$8.85 \cdot 10^{-7}$

Table 7.1: Material properties for electrostatic actuator design.

7.3.2 Optimization for low input voltage

For this first numerical example, the input voltage is set to 20 V and the available amount of quartz is limited to 30% of the design domain volume. The workpiece is modeled by a spring with a stiffness of 100 $\mu N/\mu m^2$ (as we have a 2D model with undefined thickness, the stiffness is given per micrometer of the model thickness). A continuation procedure is used for the output direction control constraint from Eq. (7.1) in order to gradually reduce the value of e from 1 to 0.01 during first 100 iterations. The optimization domain is meshed with 80 times 60 elements, which results in 4800 design variables and a density filter with a radius of 0.75 µm (i.e. 1.5 times the element

size) is used.

As noticed by Yoon and Sigmund [129], the electrostatic actuator design problem has got many local optima and the final design is very dependent from the starting point selected in the design space. Therefore the optimization problem presented in Figure 7.3 has been solved for different initial material distributions. Uniform density field and several random density distributions have been attempted. For the present application, the most efficient design obtained is presented in Figure 7.4. This design has been obtained starting from a uniform material distribution. It should be regarded as a local optimum of the optimization problem as better solutions might exist.

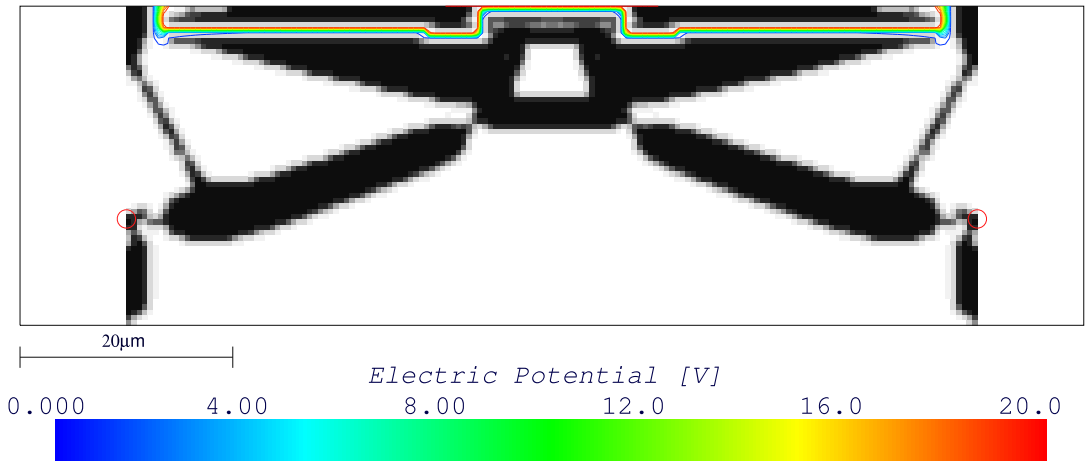


Figure 7.4: Resulting topology for a uniform initial distribution.

The actuator design presented in Figure 7.4 required 1604 iterations to converge. Both output ports are indicated by a circle in the figure. For $V_{input} = 20$ V, the output displacement at these two points is equal to $3.8 \cdot 10^{-3}$ µm (i.e. an output force $f_{out} = k \cdot u_{out} = 0.38$ µN/µm). As shown by the isovoltage curves in Figure 7.4, electrostatic effects are concentrated at the top of the domain. These curves follow the gap that separates the two electrodes on which electrostatic forces are generated. The upper electrode is directly connected to the driving voltage input point at the center of the domain. As the driving voltage is only provided on a limited part of the upper border, the purpose of this electrode is to conduct the driving voltage on both sides of the design domain. Conversely, the lower electrode is connected to ground voltage provided at the bottom of the domain through the articulated mechanism in the lower part of the domain.

The electrostatic actuator operation is illustrated in Figure 7.5 where the deformations have been amplified 400 times. As the upper electrode is fixed and does not participate to the device deformation (though it contributes to electrostatic force creation), it has been removed from the figure for the sake of clarity. The electrostatic forces pull the lower electrode up (the upper being fixed to the top of the domain). The compliant mechanism designed below the mobile electrode converts this vertical movement into a

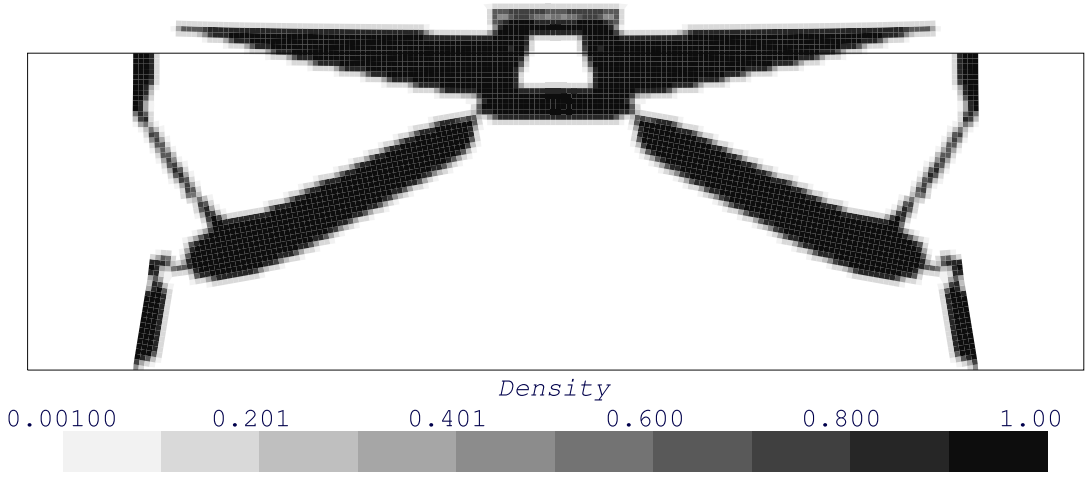


Figure 7.5: Resulting topology for a uniform initial distribution.

horizontal displacement of the two output ports towards the inside of the device.

As already mentioned, the electrostatic actuator design problem presents several local optima. These local optima can correspond to very different designs as illustrated in Figure 7.6 obtained with a random initial distribution. Like the last design, the electrode connected to driving voltage remains fixed while the ground electrode is mobile. However the operating principle of this actuator is slightly different from the previous one. The electrostatic forces are now mostly horizontal and pull the ground electrode towards the center of the actuator. This horizontal motion is transferred to the output ports through two horizontal bars. This design required 652 optimization iterations and leads to an output displacement of $1.9 \cdot 10^{-3}$ µm (i.e. $f_{out} = 0.19$ µN/µm) that is twice less than the previous design.

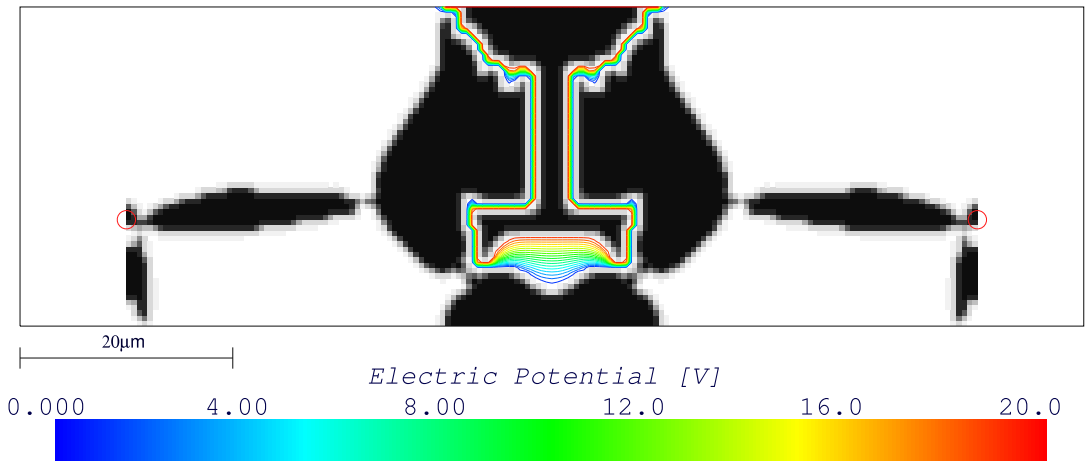


Figure 7.6: Resulting topology for a random initial distribution.

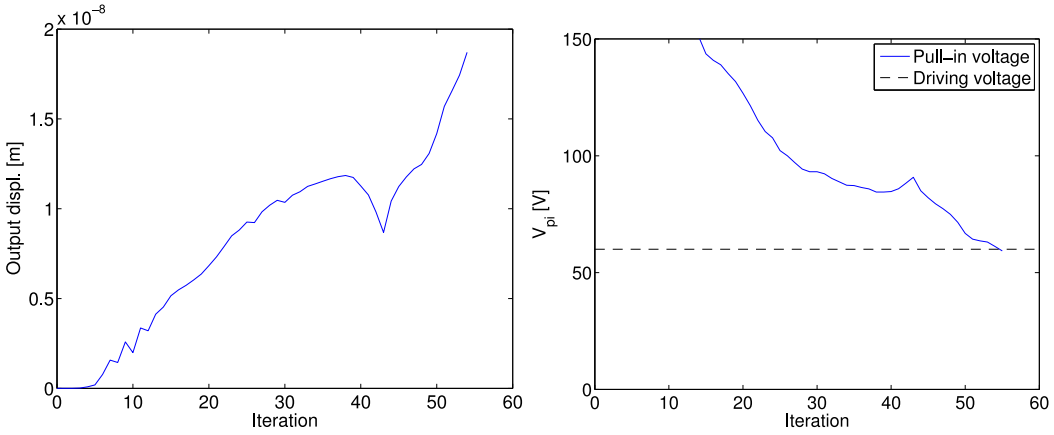
The two presented designs show that the optimization process tends to create actuators with minimum gap between electrodes. This behavior is consistent as it maximizes the

generated electrostatic force since it is proportional to the inverse of the gap. However, as noticed by Yoon and Sigmund, the use of thinner mesh leads to final design presenting smaller gaps. To avoid mesh dependency, Yoon and Sigmund proposed to use the erode filter inspired from image processing and introduced by Sigmund [100]. This filter provides good results and allows controlling heuristically the gap width of the actuator. In the present thesis, we choose to focus on the study of the necessity and the impact of including a stability constraint into the optimization problem. Therefore, the filtering approach is not implemented.

7.3.3 Optimization for high input voltage

The output displacement amplitude provided by the two designs presented in the previous section is small with respect to the device dimensions. Indeed, the actuator is several micrometers long while the output displacement magnitude is about 10^{-3} µm. To improve the output displacement, the simplest way consists in increasing the input voltage of the device to generate higher electrostatic forces.

Therefore, we attempt to solve the optimization problem presented in Section 7.3.1 with an input voltage of 60 V. However, the optimization procedure is not able to reach convergence because of the failure of the Newton-Raphson analysis after 55 iterations. Actually, at this point of the optimization process, the Newton-Raphson algorithm is unable to find the equilibrium position of the design for $V_{input} = 60$.



(a) Output displacement history.

(b) Pull-in voltage history.

Figure 7.7: History of the actuator behavior.

The origin of the problem can be explained by computing the pull-in voltage of the design at each iteration. The history of pull-in voltage corresponding to the first 55 iterations is presented in Figure 7.7(b) while Figure 7.7(a) represents the output displacement history. As expected, the output displacement increases across iterations. The drop that can be observed between iterations 40 and 50 is due to the output control constraint which becomes active from this point. Pull-in voltage follows an inverse

trend and decreases along iterations up till iteration 55 where it becomes smaller than the driving voltage (marked by the dashed line in Figure 7.7(b)). This means that, at iteration 55, there is no equilibrium position for the fixed input voltage (see Section 2.3). Therefore it explains why the Newton-Raphson procedure is unable to converge at this stage of the optimization process.

7.3.4 Physical interpretation

The previous section has shown that a too high driving voltage may cause the failure of the optimization procedure. The influence of the driving voltage on the success of the optimization procedure can be put in evidence on the basis of the simplified electromechanical actuator presented in Section 2.3. The sketch of this model is reminded in Figure 7.8. It represents a parallel plate capacitor with one plate fixed while the second is mobile and suspended by a spring.

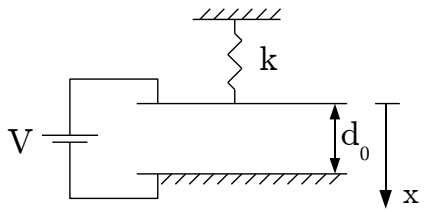


Figure 7.8: Simplified electromechanical actuator.

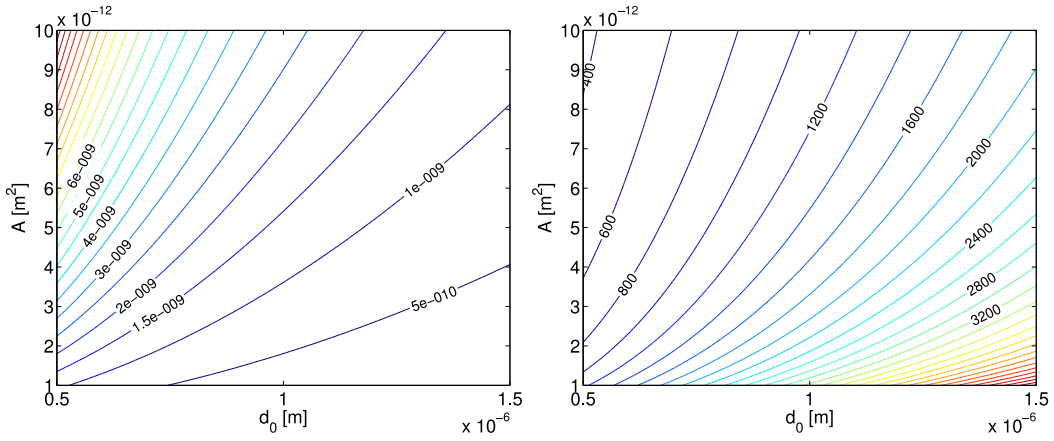
On the basis of this actuator, a two variables optimization problem similar to the one considered for topology optimization (see Eq. (7.1)) is defined. It consists in maximizing the displacement x of the mobile electrode for a given driving voltage V_{input}. The design variables of this optimization problem are the area A of the plane capacitor and the initial gap d_0. Mathematically, this optimization problem can be written as follows:

$$\max_{d_0,A} x\left(d_0, A, V_{input}\right), \quad s.t. \begin{cases} d_{0,min} \leqslant d_0 \leqslant d_{0,max}, \\ A_{min} \leqslant A \leqslant A_{max}. \end{cases}$$

For the first example, the driving voltage is fixed to 100 V and the design variables bounds are fixed as follows:

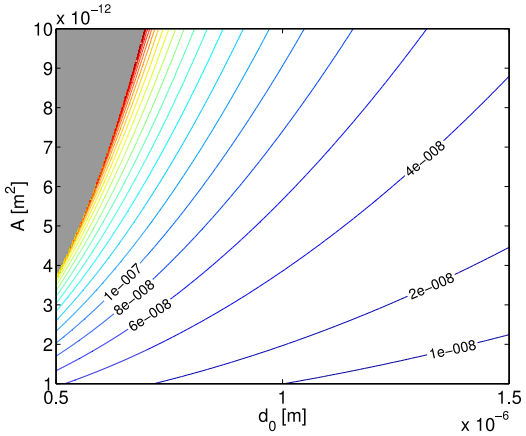
$$d_{0,min} = 0.5 \cdot 10^{-6} \text{ m}, \qquad d_{0,max} = 1.5 \cdot 10^{-6} \text{ m},$$
$$A_{min} = \cdot 10^{-12} \text{ m}^2, \qquad A_{max} = 10^{-11} \text{ m}^2.$$

The isolines of the objective function over the optimization domain are presented in Figure 7.9(a). The optimum can easily be identified in the upper left corner of this figure for the maximum electrode area A and a minimum gap d_0. Physically speaking,



(a) Output displacement isolines, $V_{input} = 100$ V.

(b) Pull-in voltage isolines.



(c) Output displacement isolines, $V_{input} = 600$ V.

Figure 7.9: Behavior of the actuator over the design domain

this combination leads to the maximum force as the area is in the numerator of Eq. (2.1) while d_0 is in the denominator.

Conversely, as illustrated in Figure 7.9(b), pull-in voltage behaves in the opposite way with respect to the output displacement. Indeed, the upper left corner of the Figure 7.9(b) corresponds to the minimum value of pull-in voltage over the optimization domain. Therefore, maximizing the output displacement leads to pull-in voltage minimization. In the present case the solution corresponding to the upper corner of the optimization domain is valid as the pull-in voltage of the optimal design is equal to 366 V which is greater than the driving voltage. Hence, in this first example, the optimal actuator is stable similarly to the results obtained in Section 7.3.2.

However, the optimization outcome may be very different if the driving voltage is increased to 600 V. In this case, the output displacement becomes theoretically infinite over one part of the design domain. This is illustrated in Figure 7.9(c) where the gray area corresponds to infinite value of the objective function. Of course, the pull-in volt-

age is independent from the input voltage and the isolines presented in Figure 7.9(b) are still valid. By comparing the shape of the gray area with the pull-in voltage isolines, we can see that the boundary of the gray area coincides with the 600 V isovalue of pull-in voltage. In other words, on the boundary of the gray area, the pull-in voltage is exactly equal to the driving voltage and inside the gray area the pull-in voltage is lower than the driving voltage.

From the optimization point of view, this situation is problematic. Indeed, in Figure 7.9(c) the isolines show that the output displacement increases monotonously in direction of the gray area. This means that a gradient based maximization would always end up in the area of the design domain where the output displacement is infinite. Therefore, the resulting design is unstable for the given driving voltage. With the optimization problem presented in Section 7.3.3, it is impossible to compute the response of the device to the driving voltage. Thus, the optimization procedure is unable to proceed and to converge to a final design.

7.3.5 Summary

As already demonstrated by Yoon and Sigmund [129], this section first confirms that design of electrostatic actuators using topology optimization is possible and leads to interesting results. Some issues as the existence of local optima and the tendency to create very small gaps (i.e. mesh dependency) noted by Yoon and Sigmund have also been observed.

Nevertheless, this conclusion only holds as long as the input voltage is chosen sufficiently small. Indeed, during optimization, the design pull-in voltage tends to decrease and may possibly become smaller than the driving voltage. As shown on basis of a numerical application, it is then impossible to compute the structural response and the optimization process has to be stopped. Moreover, the study of a simple two-variable optimization problem demonstrates that, in this case, the optimization problem lacks solution.

Therefore, it comes out that a control of the pull-in voltage along the optimization process is needed in order to prevent the pull-in voltage to become smaller than the operation voltage. The next section proposes to introduce such a stability restriction based upon the previous chapter developments.

7.4 Electrostatic actuator design including pull-in voltage constraint

7.4.1 Optimization problem

The pull-in voltage control strategy proposed in this section consists simply in adding a constraint to the optimization problem given in Eq. (7.1). This constraint imposes

a lower bound on the design pull-in voltage. In order to avoid the stability problems presented in Section 7.3, one could have simply fixed the pull-in voltage lower bound to the input voltage V_{input}. However, the optimizer we are using (*Conlin*) resorts to first order approximations of the design functions. Thus, there is no guarantee that the design update satisfies exactly the constraints. Moreover, practically speaking, it is not possible to operate the actuator exactly at pull-in voltage as it leads to a collapse of the structure and to a structural response that is not predictable in the current framework.

Therefore, a safety margin is introduced between the pull-in voltage and the input voltage by setting the lower bound on the pull-in voltage $V_{pi,min}$ greater than the driving voltage. Starting from the optimization problem written in Eq. (7.1), the new optimization problem is obtained by introducing the pull-in voltage constraint:

$$\begin{aligned} &\max_{\boldsymbol{\mu}} u_{out}\left(\boldsymbol{\mu}, V_{input}\right) \ , \\ s.t. \ &\begin{cases} V_{pi}\left(\boldsymbol{\mu}\right) \geqslant V_{pi,min} > V_{input}, & \\ \left(\frac{u_{perp}}{u_{out}}\right)^2 \leqslant e^2, & \\ v\left(\boldsymbol{\mu}\right) \leqslant \overline{v} & \text{(Optional)}, \\ \mu_{\min} \leqslant \mu_i \leqslant 1 & \forall i. \end{cases} \end{aligned} \tag{7.5}$$

7.4.2 Modifications of the optimization procedure

The introduction of the pull-in voltage constraint into the optimization problem brings an additional design function (i.e. the pull-in voltage). Consequently, the pull-in voltage and its sensitivities need to be evaluated by the optimization procedure. The related optimization procedure flowchart is presented in Figure 7.10. It consists in a combination of the one developed for pull-in voltage in Chapter 4 (see Figure 4.3) and the one proposed in Section 7.2.3 of this chapter (see Figure 7.2).

The structural analysis is now composed of two separate analyzes. The first one aims at computing the response of the design for a fixed voltage and is carried out with a Newton-Raphson analysis. The second one resorts to the normal flow algorithm in order to evaluate the pull-in conditions. The necessity for two distinct analyzes comes from the specificity of each design function; it is not possible to determine pull-in voltage using Newton-Raphson and the normal flow is not tailored to compute the response to a given voltage.

Nevertheless, the impact on the computational time of the additional constraint is high. Indeed, the computational time required by normal flow algorithm to find pull-in point can be up to 10 times higher than the computational time of the response to the input voltage using Newton-Raphson. Moreover, at the beginning of the optimization, the pull-in voltage can be very high (far more than $V_{pi,min}$) as the density distribution is mostly uniform (or still random) over the optimization domain. This results in very time consuming first pull-in searches as normal flow algorithm progresses along the equilibrium curve starting from rest position ($V = 0$ V). However, with a so high pull-in voltage, the pull-in constraint is unlikely to influence the next design update.

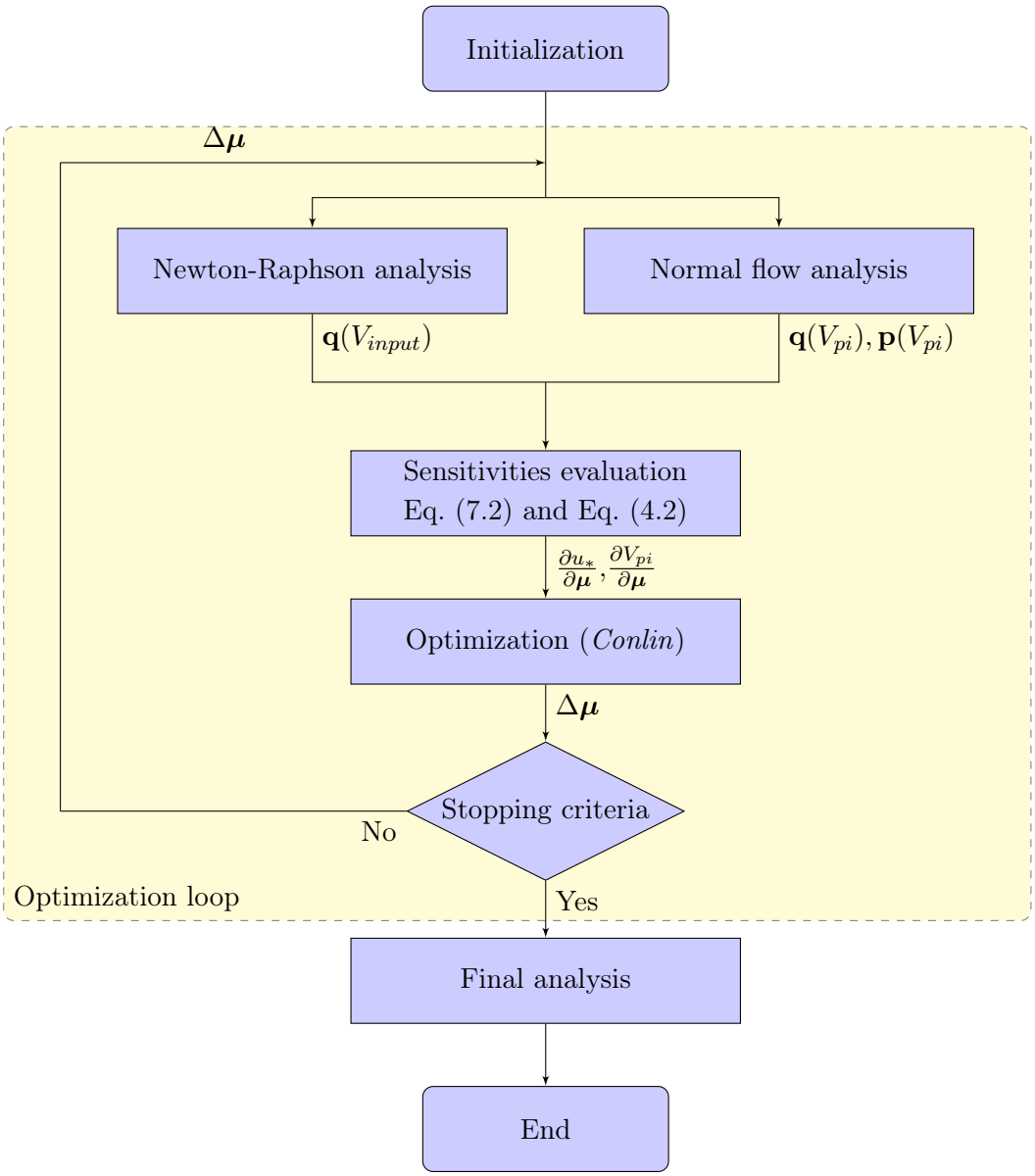


Figure 7.10: Output displacement optimization procedure flowchart.

Therefore, we choose to disable to disable the pull-in voltage constraint if, during the normal flow step, the voltage becomes significantly larger than its lower bound $V_{pi,min}$ (for instance more than two times greater as used in next application). The pull-in search is then aborted and the pull-in voltage constraint is removed from the optimization problem for the current iteration. This strategy is very similar to the active constraint selection strategy which has been proposed for topology optimization problems including stress constraints [35]. The constraints which have very little chance to become active during current optimization iteration are removed from the optimization problem in order to save computational time.

7.4.3 Numerical application

In order to demonstrate the efficiency of the proposed approach, the numerical application which failed to converge in Section 7.3.3 because of a too high input voltage is revisited with a pull-in voltage constraint. The safety margin between the requested minimum pull-in voltage and the input voltage is set to 10% of the input voltage, i.e. $V_{pi,min} = 66$ V. As mentioned in last section, in order to save computational time, the pull-in constraint is only enabled when the pull-in voltage is smaller than two times its lower bound. The geometry of the optimization problem as well as the optimization parameters remain the same as the one presented in Section 7.3.3.

The most efficient design is obtained this time for a random initial distribution; it is presented in Figure 7.11. The operation principle of this design is similar to the second design obtained with a lower driving voltage (see Figure 7.6). The two ground electrodes are pulled towards the center of the device and this displacement is transmitted to the output ports by a very simple mechanism.

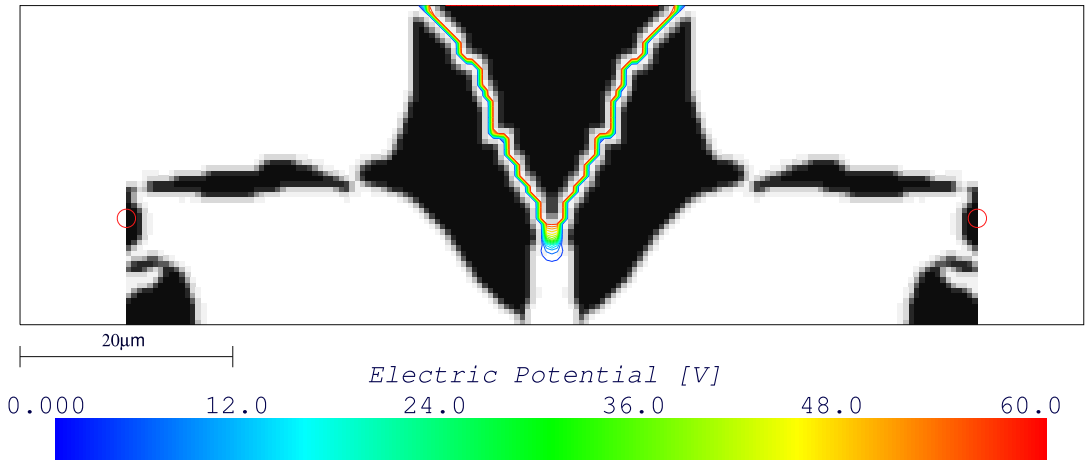
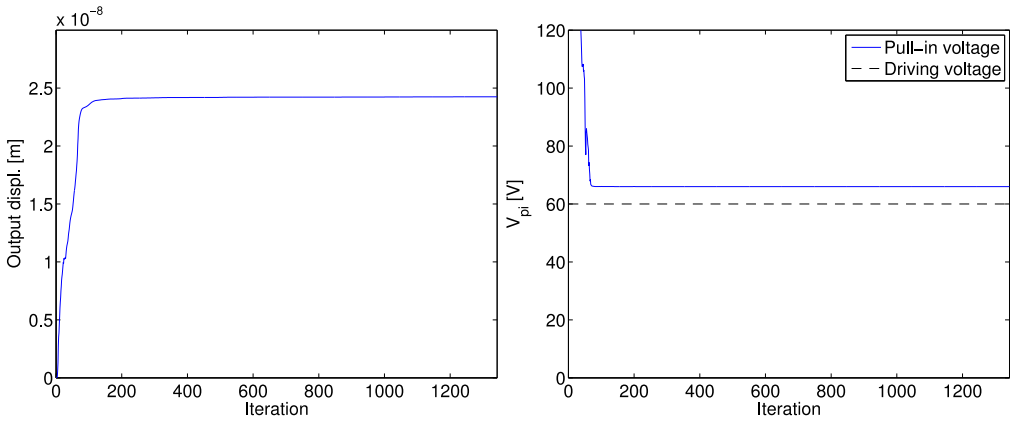


Figure 7.11: Resulting topology with a pull-in voltage constraint.

Thanks to the input voltage increase, it is possible to reach higher output displacement. Indeed, the final output displacement is equal to $2.4 \cdot 10^{-2}$ µm (i.e. $f_{out} = 2.4$ µN/µm). That is to say 6 times more than the output displacement obtained with a driving

voltage of 20 V. The history of the output displacement is presented in Figure 7.12(a). Within 200 iterations, the objective function has almost reached its final value and the design is very similar to the final one. However, slow design modifications and small oscillations continue to occur after the first 200 iterations. In total, 1343 iterations are needed to satisfy the stopping criterion (maximum pseudo-density variation lower than $2 \cdot 10^{-3}$).

The value of pull-in voltage along optimization is plotted in Figure 7.12(b). At the beginning of the optimization process, the pull-in constraint is disabled as pull-in voltage is larger than two times its lower bound. It is enabled at iteration 33 and remains enabled until the end of the optimization process. After being enabled the constraint becomes very rapidly active as the pull-in voltage decreases up to its lower bound within a few iterations. Then, the constraint prevents the pull-in voltage from decreasing below $V_{pi,min}$ and ensures the stability of the design.



(a) Output displacement history.

(b) Pull-in voltage history.

Figure 7.12: History of the actuator behavior.

7.5 Influence of large displacement modeling

In the designs presented above, the optimized distribution of material resorts to so called *compliant mechanisms* in order to transfer the force from the electrode to the output ports. As defined by Sigmund [98], a *compliant mechanism* is a mechanism that gains its mobility from the flexibility of some or all of its members. However, as mentioned at the beginning of the present chapter, C. Pedersen *et al.* [82] show that topology optimization of compliant mechanisms using linear mechanical modeling (small displacement hypothesis) may lead to results that do not reach expected performance in practice. C. Pedersen *et al.* explain this issue by the unability of small displacement analysis to predict locking of the compliant mechanism. In order to avoid such problems, geometric nonlinearity (large displacement hypothesis) has to be taken into account.

Therefore, the need to consider large displacement hypothesis is investigated in this

section. Electromechanical finite elements which include large displacement modeling were already available in our software (see Ref. [90]). These elements have been adapted to take fictitious permittivity and electrostatic force filtering into account. Except for these modifications, no further changes are necessary since the modeling procedure and the sensitivity analysis are already able to tackle with nonlinearity coming from the electrostatic effects.

7.5.1 Mechanically nonlinear post-analysis of previous designs

To assess the influence of large displacement hypothesis on the optimization outcome, we first test its impact on the performance of previous designs optimized with small displacement modeling. Large displacement post-analysis is carried out on two designs:

1. The actuator design obtained for $V_{input} = 20$ V without pull-in constraint and starting from a uniform distribution (see Figure 7.4),
2. The actuator optimized for $V_{input} = 60$ V including a pull-in voltage constraint (see Figure 7.11).

The resulting output displacements are presented in Table 7.2 together with the original objective function value. Because of the small magnitude of the deformations for the first design ($V_{input} = 20$ V), large displacement modeling has little impact on the behavior of this design. Conversely, the difference is more significant for the second design ($V_{input} = 60$ V) due to the larger deformation of the device during operation. For the second design, the large displacement hypothesis results in an increase of the output displacement of 19%. However the structural pull-in voltage decreases from 66 V to 62.4 V, which means that the pull-in voltage constraint imposed during design is violated.

Problem formulation	Small displ.	Large displ.
$V_{input} = 20$ V, Figure 7.4	$3.78 \cdot 10^{-3}$ µm	$3.83 \cdot 10^{-3}$ µm
$V_{input} = 60$ V, Figure 7.11	$2.42 \cdot 10^{-2}$ µm	$2.87 \cdot 10^{-2}$ µm

Table 7.2: Influence of the modeling hypothesis on the performance of previous designs

C. Pedersen *et al.* [82] observed that nonlinear analysis of designs obtained with small displacement hypothesis generally lead to a lower performance due to the locking of the mechanism. This effect does not appear here. Instead, large displacement analysis leads to a better performance which might be surprising since the actuator was optimized under a small displacement hypothesis. This improvement results from the large rotations taken into account by large displacement analysis while they are neglected by small displacement hypothesis. Depending on the circumstances, large rotations can lead to a higher or a lower performance. For instance, a numerical experiment consisting in reversing the direction of the electrostatic forces has been carried out on the

second design. Of course, the output displacement is then oriented towards the outside but its magnitude is now lower with the large displacement analysis ($2.22 \cdot 10^{-2}$ µm) than with a small displacement analysis (still $2.42 \cdot 10^{-2}$ µm) for $V_{input} = 60$ V. If the voltage is decreased, both analyzes converge towards the same values as the effect of large rotations becomes negligible. Therefore, the higher performance obtained with large displacement analysis in Table 7.2 can be seen as a coincidence.

7.5.2 Actuator optimization including geometric nonlinearity

According to the results in Table 7.2, it is interesting to solve the optimization problem that gives rise to the second design considering geometric nonlinearity. The optimization problem given in Eq. (7.5) is then solved in the same conditions than in Section 7.4.3 but with large displacement modeling hypothesis. Numerical instabilities occurring during the geometrically nonlinear analysis were reported by C. Pedersen *et al.* [82] and Bruns and Tortorelli [25]. However, these issues have not been observed in the present optimization problem. The resulting design is presented in Figure 7.13. The new design is only slightly different from the previous one (see Figure 7.11).

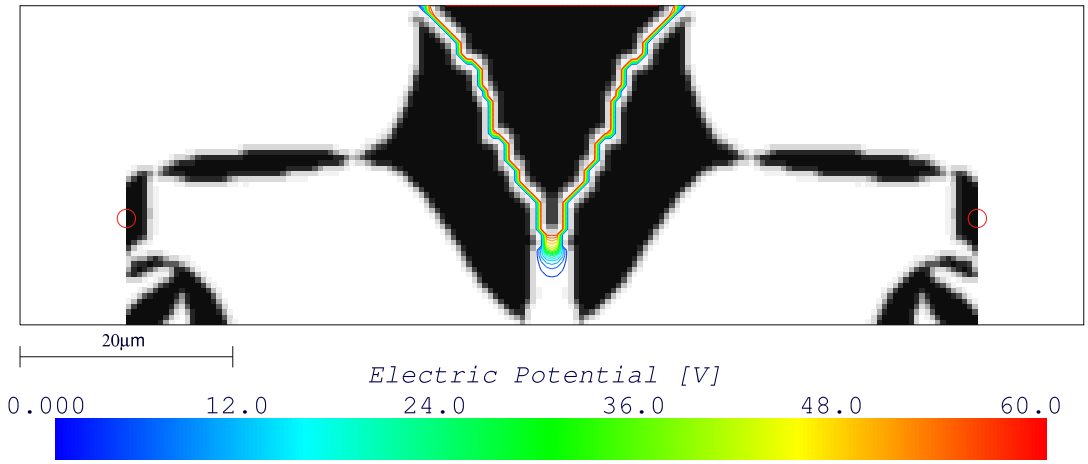


Figure 7.13: Resulting topology with a pull-in voltage constraint and large displacement modeling.

However, from the point of view of the performance, differences can be noticed with respect to previous results. First of all, this design satisfies the pull-in voltage constraint while it is not the case of previous design if it is post-analyzed with large displacement hypothesis. This comes at the cost of a lower output displacement as the output displacement provided by the new design is equal to $2.64 \cdot 10^{-2}$ µm (i.e. $f_{out} = 2.64$ µN/µm) against $2.87 \cdot 10^{-2}$ µm for large displacement analysis of previous design. Nevertheless, the output displacement obtained here is still 9% larger than the value reached previously with linear mechanical modeling.

Figure 7.14 shows the influence of modeling hypothesis on the history of the objective function. The two curves coincide during the first iterations when displacement mag-

nitude remains small. After a few iterations, while the objective function is increasing, the two curves separate from each other. The geometric nonlinear analysis reaches its final value after 1737 iterations which shows that the optimization process may require a large number of iterations to converge.

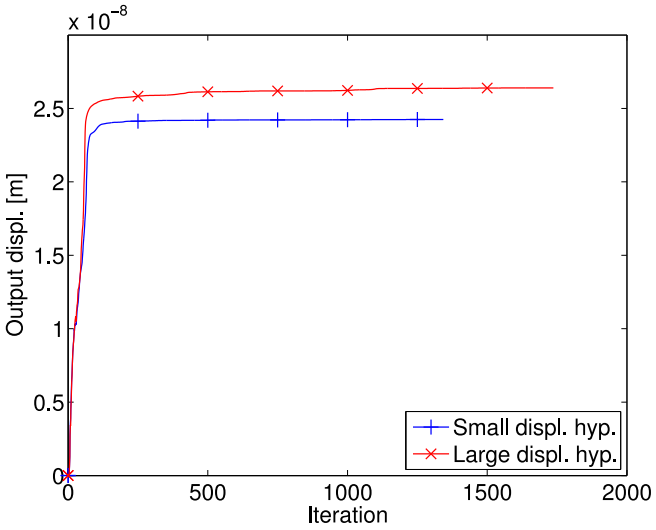


Figure 7.14: Output displacement history.

7.6 Conclusion

The present chapter extends the research proposed by Yoon and Sigmund about electrostatic actuator optimization [129] by introducing pull-in voltage constraint and considering large displacement analysis. Starting from the electrostatic actuator optimization problem proposed by Yoon and Sigmund, we have shown that, while it provides good results for low input voltage, the use of higher driving voltage may result in stability issues and failure of the optimization process. Indeed, as illustrated on a simple problem, the maximization of the output displacement may lead the optimizer to decrease the pull-in voltage of the actuator such that the structural pull-in voltage becomes smaller than the input voltage.

Therefore, a pull-in voltage constraint is introduced in the optimization problem in order to impose a minimum pull-in voltage level and to ensure the design stability. The pull-in voltage constraint takes advantage of the developments of previous chapters and demonstrates the usefulness of pull-in voltage function in MEMS design. As illustrated with a numerical application, thanks to this new constraint, we are now able to solve design problems for which the original optimization procedure does not reach solution.

As a first approach, optimization has been carried out considering small displacement assumption. To verify the validity of this hypothesis, a large displacement post-analysis has been performed on the resulting design. It turns out that a nonlinear post-analysis

leads to significant behavior differences in terms of output displacement and of pull-in voltage for the design that provides the largest output displacement. Fortunately, with large displacement hypothesis, the device is still stable if the input voltage is applied but the pull-in voltage has decreased and the pull-in constraint is not satisfied anymore. Therefore, the corresponding design problem has been solved with large displacement hypothesis. The resulting actuator is not very different from the one designed considering small displacement, but the new design satisfies the pull-in voltage constraint.

Chapter 8

Conclusion

8.1 Summary of the work

Topology optimization of electrostatic microsystems is a challenging task. Indeed, firstly, the behavior of these microsystems is relatively complex as it involves two physical fields and leads to a nonlinear response with a limit point and maybe some bifurcation points. The existence of the limit point is inherent to electrostatic actuation and results in an unstable behavior of the device called pull-in effect. In this work we show that this behavior has to be included into the optimization procedure in order to guarantee the stability of the final design and to ensure convergence of the optimization process. The second difficulty for optimization of electromechanical microdevices lies into the design dependency of electrostatic forces. Literature about topology optimization problems including design dependent loads reports difficulties caused by local instability modes and recommends a careful definition of the material model. This situation has also clearly been observed in this work.

In order to consider the difficulties one by one, we chose to first focus on the development of a topology optimization procedure able to control the stability of electromechanical microdevices. Therefore, a simplified version of the optimization problem which allows limiting the design dependence of electrostatic forces is firstly considered. In this intermediate problem, an imposed interface is defined between the optimization and the electrostatic domains. As a result, the application surface of electrostatic forces is fixed and only their magnitude is design dependent.

A first approach to control electrostatic microsystems stability during design consists in directly considering the pull-in voltage as a design function of the optimization problem. The ability to evaluate the pull-in voltage and its sensitivities is implemented in the framework of the simplified optimization problem. A design problem consisting in maximizing the pull-in voltage illustrates the good convergence properties of the developed method. Moreover, the addition of a design constraint allowed improving the manufacturability of the optimal results.

Alternatively, rather than evaluating exactly the value of pull-in voltage, it is also possible to get an estimated value of the instability voltage by formulating a stability eigenproblem at a stable point of the equilibrium path. The closer the eigenproblem is formulated from pull-in point, the more accurate is the estimation. The great advantage of this approach is that it allows capturing the pull-in instability mode as well as several higher order instability modes while the use of pull-in voltage design function only allows detecting the first instability mode. For each estimated instability mode an estimated pull-in voltage can be computed. Therefore, the optimization problem aiming at maximizing pull-in voltage is reformulated by including several estimated pull-in voltages into a multi-objective optimization problem. When used as objective functions, the estimated pull-in voltage provides great results as design updates actually lead to pull-in voltage improvement. Thanks to this approach, it is possible to avoid design oscillations that can appear with the pull-in voltage design function. However, this comes at the cost of a computationally rather expensive sensitivity analysis. Moreover, as shown on numerical applications, the approach requires the implementation of an appropriate treatment for repeated eigenvalues.

Next, the development of an optimization procedure able to control pull-in voltage is carried on by removing the imposed interface separating the optimization domain from the electrostatic domain. This leads us to the so called 'generalized' optimization problem where the electrostatic forces location, direction and intensity become design dependent. Unfortunately, this generalization has not been possible for the approach based on the stability eigenproblem. Indeed, the generalization of the optimization problem leads to an asymmetric tangent stiffness matrix, which is unsuited to the formulation of a stability eigenproblem. Instead, a generalized optimization problem involving pull-in design function is established.

In spite of the precautions taken in the definition of the material model, the first numerical application fails to converge because of local modes. After investigation, it comes out that the observed local modes result from the inaccuracy of electrostatic forces. As the enrichment of the finite elements does not solve completely the problem, we choose to resort to a force filtering technique which removes forces on nodes surrounded by void elements. Thanks to the force filter, the optimization process can reach convergence as illustrated by the proposed numerical applications. Moreover, in the context of the generalized problem, a comparison of maximum pull-in voltage design with linear compliance optimization shows that the two approaches provide significantly different results.

Finally, the usefulness of pull-in voltage design function is demonstrated on the basis of the electrostatic actuator design problem. The objective of actuator design is to find the design maximizing the output displacement for a given input voltage. While the original optimization problem proposed in literature provides good results for low input voltages, it may fail if the input voltage is increased. This issue can be avoided by including a pull-in voltage constraint into the optimization problem. Indeed, as explained on a two-variable optimization problem, the original actuator design problem has a natural

tendency to reduce pull-in voltage. The purpose of the pull-in constraint is to prevent the design pull-in voltage from decreasing below the prescribed input voltage. Thanks to the developed pull-in function, the design procedure is able to converge independently from the selected input voltage. Moreover, the influence of large displacement analysis is also investigated. Even if the output displacement provided by the optimized designs is still moderated, a significant difference in terms of performance has been noticed.

8.2 Achievements of the thesis

The final objective of the present thesis is to develop a topology optimization procedure able to control pull-in voltage. This optimization procedure is developed gradually by considering difficulties one after the other. Firstly, the development of an optimization criterion based on the pull-in voltage leads us to the following achievements:

- Development of a path following procedure based on normal flow algorithm able to compute accurately the pull-in configuration. The normal flow algorithm has been implemented into our finite element software and has been adapted in order to locate accurately the pull-in point. This procedure is essential for the sensitivity analysis of pull-in voltage.

- Computation of pull-in voltage sensitivities in the framework of a monolithic finite element formulation. The semi-analytic sensitivity expression enables using gradient based optimizer.

- Treatment of the local pull-in modes related to finite element inaccuracies using a force filtering strategy. The force filter removes artificial forces appearing on nodes surrounded by void elements.

- Illustration of the practical interest of pull-in voltage design function by introducing the new design function as a constraint into the topology optimization problem of an electrostatic actuator.

Additionally, another approach to control pull-in instability is proposed. Conversely to the method consisting in computing the actual pull-in voltage, this method resorts to a stability eigenproblem. In this framework, the principal contributions are:

- Development of an eigenproblem approximation allowing estimating actual pull-in voltage and higher order instability modes. Because the electromechanical tangent stiffness matrix is indefinite, several eigenproblem formulation are tested. The retained eigenproblem is based on a condensed version of the coupled tangent stiffness matrix in order to get a positive semi-definite matrix.

- Computation of the eigenvalues sensitivities using a semi-analytic approach. In order to avoid finite difference involving the global tangent stiffness matrix, the

ability to compute analytically its variations with respect to infinitesimal perturbations of the generalized displacements is developed and implemented. This improvement allows cutting by 30% the sensitivities computational time and provides more accurate sensitivities with respect to finite differences.

- Formulation of a multi-objective optimization problem based on the estimated pull-in voltages provided by the stability eigenproblem. Thanks to the higher order instability modes captured by the stability eigenproblem, it is possible to avoid design oscillations arising with pull-in voltage optimization. Moreover, dedicated treatment of repeated eigenvalues is also developed and implemented.

The developments proposed along this thesis have been implemented into *Oofelie* software package.

8.3 Future works

8.3.1 Extension to three-dimensional problems

All topology optimization numerical applications proposed in the present thesis are two-dimensional, such that the computational time remains reasonable. However, the extension of the developed optimization procedures to three-dimensional applications is of course possible. Hexahedral electromechanical elements are available in *Oofelie* and can be adapted to take fictitious permittivity and force filtering into account. Thanks to this improvement, other optimization problems could be considered as the one suggested in next section.

8.3.2 Suspension design for electrostatic oscillators

Some RF-filters and the gyroscope presented in Chapter 2 are based on oscillations of a proof mass in a plane parallel to the substrate. However, because of the application of a bias voltage, the proof mass can also collapse onto the substrate if its suspensions are not stiff enough to avoid pull-in effect. Therefore, one can formulate an optimization problem consisting in designing the proof mass suspension springs in order to reach a prescribed in-plane resonance frequency while satisfying a minimum out-of-plane pull-in voltage.

8.3.3 Introduction of additional design constraints

The optimization problems studied along this thesis can be enriched by introducing other design constraints in order to improve the realism of the designs. Stress constraints [24] could be added to the optimization problem in order to ensure that the optimal structure can resist to the electrostatic load.

Moreover, the development of other manufacturing constraints is also of great interest. For instance, suspended MEMS structures are fabricated by firstly creating on the substrate a sacrificial layer. Next, the structural layer is deposited on top of the sacrificial layer. To release the structure from the substrate, the sacrificial layer is then removed using acid. However, as illustrated in Figure 8.1, if the structure covers a large area, regularly spaced holes need to be created into it. These holes provide additional access to the sacrificial layer for the acid and accelerate the release process. Therefore, in the context of a topology optimization problem, a design constraint leading to the systematic creation of such holes would be very helpful.

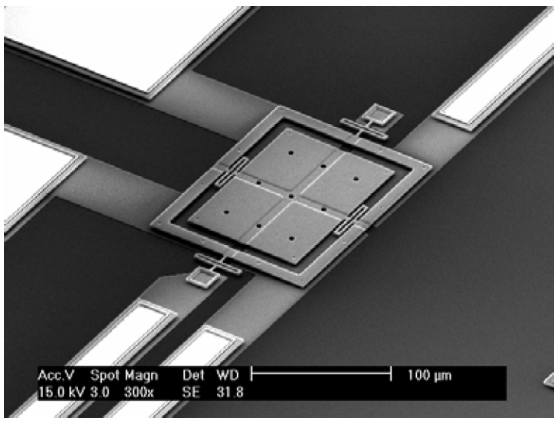


Figure 8.1: Example of suspended structure, the holes in the suspended plate accelerate the removal of the sacrificial layer[2].

8.3.4 Reliability-based optimization

MEMS devices are submitted to uncertainties on their dimensions and material properties. As we have seen, the electrostatic force is very sensitive to the electrostatic gap size. For an electrostatic actuator, a too large gap can lead to lower output displacement while a too small gap may lead to pull-in effect. An approach similar to the one proposed by Sigmund in [101] or Qian and Sigmund in [84] can be applied in order to design more robust actuators with respect to manufacturing tolerances. Moreover, variations of the material properties can be taken into account by reliability-based design as proposed by Maute and Frangopol [70].

8.3.5 Larger gaps thanks to the pull-in voltage constraint

As Yoon and Sigmund [129], we noticed the tendency of the actuator design problem to create very small (one element wide) electrostatic gaps. This leads to mesh dependency of the designs and raises concerns about the quality of the finite element analysis solution on these very thin gaps. Yoon and Sigmund solve the problem using a filtering

[2]Source: Institut d'Electronique Fondamentale (IEF), Paris

technique which allows imposing a minimum gap size. This solution makes sense from the manufacturing point of view and improves modeling precision over the gaps but prevents the optimizer from selecting the optimal gap size.

A closer look at the two-variable design problem considered in Chapter 7 suggests that the size of the gap may result from the pull-in voltage constraint. Figure 8.2 presents the contour plot of the output displacement provided by the simple actuator from Section 7.3.4. Moreover, the design domain restriction resulting from a pull-in voltage constraint $V_{pi} \geqslant 1000\,\mathrm{V}$ is indicated in Figure 8.2 by the bold black curve (the admissible domain is on the right side). As one can see the constrained optimum (marked by a circle) does not correspond to the minimum gap. As a result, in presence of a pull-in constraint, the optimum gap may be mesh independent.

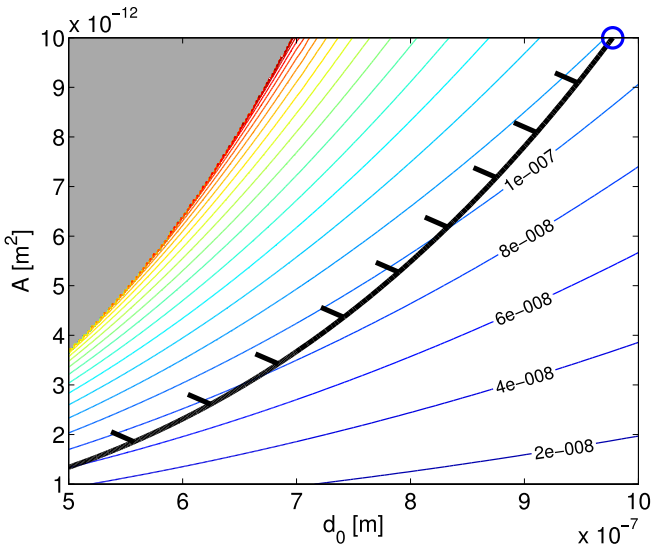


Figure 8.2: Output displacement for the two-variable design problem (see Section 7.3.4).

8.3.6 Generalization of the stability eigenproblem approach

Design oscillations due to sudden modifications of the instability mode shape have only been presented in the framework of the simplified optimization problem (i.e. with the separation between electrostatic and design domains). However, this issue can also arise when considering the generalized optimization problem. The application of the stability eigenproblem method to the generalized optimization problem would lead to a more robust optimization procedure. This improvement has not been possible in the present framework because the modeling procedure modifications (fictitious permittivity and force filtering) make the tangent stiffness matrix asymmetric. As a result, the stability eigenvalues can be complex numbers whose meaning is difficult to interpret. Therefore, one should consider other modeling options in order to avoid those artificial modifications of the modeling procedure.

Improving the accuracy of electrostatic forces

The force filtering aims at removing artificial local modes in void areas of the optimization domain. One possibility is to replace this approach by an enrichment of the finite elements even though this approach has not been successful for the maximum pull-in voltage optimization problem. Indeed, if a pull-in voltage constrained problem is considered, we only need pull-in of local modes to be larger than the voltage range of interest. Therefore, enrichment might be enough to increase sufficiently the pull-in voltage of artificial modes. For this purpose, special shape functions which ensure the continuity of the derivative across element boundaries could be used (see Refs [63, 64]).

Electric conduction modeling

The fictitious permittivity is introduced in order to obtain perfectly conducting behavior on the structural part of the optimization domain. Instead, an electrodynamic formulation [91] could be investigated in order to rigorously model both electric conduction and electrostatic effects. However, it comes at the price of a non-elliptic modeling problem.

Appendix A

Semi-analytic derivative of the tangent stiffness matrix

A.1 Introduction

This appendix develops the semi-analytical expressions of the tangent stiffness matrix sensitivities relative to generalized displacements. The finite element formulation used in the present thesis has been developed by Rochus in Ref. [90, 91]. The approach proposed by Rochus results in a tangent stiffness matrix where four main blocks can be identified:

$$\mathbf{K}_t = \begin{bmatrix} \mathbf{K}_{\phi\phi} & \mathbf{K}_{\phi u} \\ \mathbf{K}_{u\phi} & \mathbf{K}_{uu} \end{bmatrix}$$

$\mathbf{K}_{\phi\phi}$ is the generalized stiffness linking electrostatic charges to electrostatic potentials, $\mathbf{K}_{uu}$ is the tangent stiffness related to mechanical degrees of freedom while $\mathbf{K}_{\phi u}$ and $\mathbf{K}_{u\phi}$ are the coupling matrices between the two physical fields. For the two last blocks we have, $\mathbf{K}_{\phi u} = \mathbf{K}_{u\phi}^T$.

Each block of $\mathbf{K}_t$ possesses its own mathematical definition. That's why the three next sections are each dedicated to one of these blocks and present the computation of the semi-analytic sensitivities.

The semi-analytic sensitivities are established by adapting the approach proposed by Van Miegroet in Ref. [111]. Computations are performed for the 2D case but they can of course be extended to 3D.

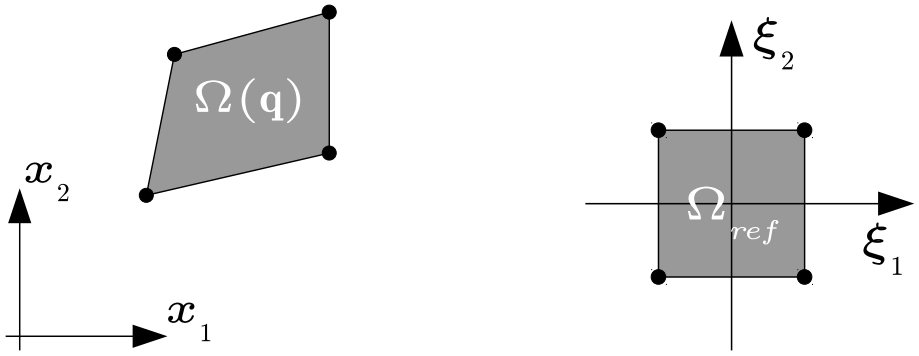
A.2 Electrostatic tangent stiffness $\mathbf{K}_{\phi\phi}$

According to Rochus in Ref. [91] the analytical expression of $\mathbf{K}_{\phi\phi}$ is:

$$\mathbf{K}_{\phi\phi} = -\int_{\Omega(\mathbf{q})} \mathbf{B}_\phi^T \boldsymbol{\varepsilon} \mathbf{B}_\phi \, d\Omega \tag{A.1}$$

In this definition, $\mathbf{B}_\phi$ is the derivation operator that converts the electrostatic potential field into its gradient and $\boldsymbol{\varepsilon}$ is the dielectric tensor. According to this definition, $\mathbf{K}_{\phi\phi}$ depends on the generalized displacements in two ways. Firstly through the operator $\mathbf{B}_\phi$ which is expressed in the (mechanically) deformed referential. Secondly, the integration domain also depends on generalized displacements. Indeed, as described by Rochus, contributions of the electrostatic energy to tangent stiffness have to be integrated into the deformed configuration as reminded in section 2.4.1.

To derive with respect to the integration domain, one can use a variable substitution to map the deformed domain $\Omega(\mathbf{q})$ onto a reference domain Ω_{ref} which is independent of the deformation. As presented in Figure A.1(a), the deformed configuration is expressed in the (x_1, x_2) referential. The coordinates of a node in this referential are equal to the sum of its position in the real space and of its displacement. The reference element (Figure A.1(b)) is defined in a fictitious space (ξ_1, ξ_2) were the nodes of the element have fixed coordinates.



(a) Deformed configuration (b) Reference space

Figure A.1: Reference and deformed configurations

The variable substitution $\mathbf{x}(\boldsymbol{\xi})$ allows passing from the reference space to real space. The Jacobian matrix $\mathbf{J}$ of this transformation is defined as:

$$[\mathbf{J}]_{i,j} = \frac{\partial x_j}{\partial \xi_i} \tag{A.2}$$

In the isoparametric finite element context, the mapping between reference and deformed configuration is defined by (using Einstein summation convention):

$$\mathbf{x}(\boldsymbol{\xi}) = N_k(\boldsymbol{\xi})\,\mathbf{X}_k \tag{A.3}$$

Where $\mathbf{X}_k$ is the deformed position of node k and $N_k(\boldsymbol{\xi})$ is the associated shape function. By using the Jacobian matrix $\mathbf{J}$ of $\mathbf{x}(\boldsymbol{\xi})$, Eq. (A.1) can be rewritten:

$$\mathbf{K}_{\phi\phi} = -\int_{\Omega_{ref}} \mathbf{B}_\phi^T \boldsymbol{\varepsilon} \mathbf{B}_\phi \, \|\mathbf{J}\| \; d\Omega_{ref}$$

Therefore the partial derivative of $\mathbf{K}_{\phi\phi}$ with respect to $\mathbf{q}$ is:

$$\frac{\partial \mathbf{K}_{\phi\phi}}{\partial \mathbf{q}} = -\int_{\Omega_{ref}} \frac{\partial \mathbf{B}_\phi^T}{\partial \mathbf{q}} \boldsymbol{\varepsilon} \mathbf{B}_\phi \left\|\mathbf{J}\right\| + \mathbf{B}_\phi^T \boldsymbol{\varepsilon} \frac{\partial \mathbf{B}_\phi}{\partial \mathbf{q}} \left\|\mathbf{J}\right\| + \mathbf{B}_\phi^T \boldsymbol{\varepsilon} \mathbf{B}_\phi \frac{\partial \left\|\mathbf{J}\right\|}{\partial \mathbf{q}} \, d\Omega_{ref}$$

A.2.1 Derivative of $\mathbf{B}_\phi$ matrix

The derivation operator $\mathbf{B}_\phi$ is defined on the basis of the gradient of shape functions of the finite element. If N_j is the shape function associated with DOF j, we have into the real (deformed) space:

$$\left[\mathbf{B}_\phi\right]_{i,j} = \frac{\partial N_j}{\partial x_i}$$

Which can be transformed into the reference space by using the definition of the Jacobian matrix Eq. (A.2):

$$\left[\mathbf{B}_\phi\right]_{i,j} = \frac{\partial N_j}{\partial \xi_k} \frac{\partial \xi_k}{\partial x_i} = \left[\mathbf{J}^{-1}\right]_{ik} \frac{\partial N_j}{\partial \xi_k}$$

When this last expression is derived with respect to generalized displacements, we get:

$$\frac{\partial \left[\mathbf{B}_\phi\right]_{i,j}}{\partial \mathbf{q}} = \frac{\partial \left[\mathbf{J}^{-1}\right]_{ik}}{\partial \mathbf{q}} \frac{\partial N_j}{\partial \xi_k}$$

Since shape functions are independent from generalized displacements, it is only needed to derive the inverse of the Jacobian matrix. This is a rather cumbersome derivative, but fortunately, it can be easily transformed into an easier expression by noting that:

$$0 = \frac{\partial \mathbf{J}\mathbf{J}^{-1}}{\partial \mathbf{q}} = \frac{\partial \mathbf{J}}{\partial \mathbf{q}} \mathbf{J}^{-1} + \mathbf{J} \frac{\partial \mathbf{J}^{-1}}{\partial \mathbf{q}}$$

So that we have:

$$\frac{\partial \mathbf{J}^{-1}}{\partial \mathbf{q}} = -\mathbf{J}^{-1} \frac{\partial \mathbf{J}}{\partial \mathbf{q}} \mathbf{J}^{-1}$$

Which means that the derivative of the inverse of the Jacobian can be computed on basis of the Jacobian matrix derivative. Evaluation of this last one is the topic of next section.

A.2.2 Derivative of the Jacobian matrix and of its determinant

Considering the isoparametric mapping introduced in Eq. (A.3), the definition of the Jacobian matrix (Eq. (A.2)) can be transformed:

$$\left[\mathbf{J}\right]_{i,j} = \frac{\partial N_k}{\partial \xi_i} X_{k,j}$$

With $X_{k,i}$ being the absolute position (including displacement in the present case) of node k along axis j. Deriving this expression with respect to $\mathbf{q}$ leads to:

$$\frac{\partial \left[\mathbf{J}\right]_{i,j}}{\partial \mathbf{q}} = \frac{\partial N_k}{\partial \xi_i} \frac{\partial X_{k,j}}{\partial \mathbf{q}}$$

As shape functions expressed in reference space are independent from the generalized displacements. The derivative $\partial X_{k,j}/\partial \mathbf{q}$ is straightforward since $X_{k,j}$ corresponds to one single element of $\mathbf{q}$:

$$\frac{\partial X_{k,j}}{\partial \mathbf{q}_i} = \begin{cases} 1 \text{ if DOF } (X_{k,j}) \equiv \text{DOF } (\mathbf{q}_i) \\ 0 \text{ if DOF } (X_{k,j}) \neq \text{DOF } (\mathbf{q}_i) \end{cases} \tag{A.4}$$

In other words, the derivative is equal to one if $X_{k,j}$ and $\mathbf{q}_i$ correspond to the same degree of freedom and equal to 0 otherwise.

In the context of section 5.4.3 where one wants to compute $\frac{\partial \mathbf{K}_t}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l}$, the derivative of $[\mathbf{J}]_{i,j}$ takes the following form:

$$\frac{\partial [\mathbf{J}]_{i,j}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l} = \frac{\partial N_k}{\partial \xi_i} \frac{\partial X_{k,j}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l}$$

By taking Eq. (A.4) into account, this last expression can be reformulated as:

$$\frac{\partial [\mathbf{J}]_{i,j}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l} = \frac{\partial N_k}{\partial \xi_i} X'_{k,j} \tag{A.5}$$

Where $X'_{k,j}$ is equal to the value of $d\mathbf{q}/d\mu_l$ component corresponding to $X_{k,j}$. Therefore, evaluating Eq. (A.5) reduces to computing $\mathbf{J}$ with a special displacement field equal to $d\mathbf{q}/d\mu_l$.

Once the derivative of the Jacobian matrix is known, the derivative of its determinant is very simple to compute. For instance, in the 2D case, the determinant of $\mathbf{J}$ takes the form:

$$\|\mathbf{J}\| = \frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_1}{\partial \xi_2} \frac{\partial x_2}{\partial \xi_1}$$

And therefore, the derivative of this determinant with respect to $\mathbf{q}$ basically involves the derivative of the elements of $\mathbf{J}$:

$$\frac{\partial \|\mathbf{J}\|}{\partial \mathbf{q}} = \frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial x_1}{\partial \xi_1} \right) \frac{\partial x_2}{\partial \xi_2} + \frac{\partial x_1}{\partial \xi_1} \frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial x_2}{\partial \xi_2} \right) - \frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial x_1}{\partial \xi_2} \right) \frac{\partial x_2}{\partial \xi_1} - \frac{\partial x_1}{\partial \xi_2} \frac{\partial}{\partial \mathbf{q}} \left(\frac{\partial x_2}{\partial \xi_1} \right)$$

A.3 Mechanical tangent stiffness $\mathbf{K}_{uu}$

Under the small displacement assumption (linear mechanics), the expression of the mechanical stiffness block of the tangent stiffness matrix is:

$$\mathbf{K}_{uu} = \int_{\Omega} \mathbf{B}_u^T \mathbf{H} \mathbf{B}_u \, d\Omega - \frac{1}{2} \int_{\Omega(\mathbf{q})} \mathbf{B}_u^T \mathbf{F}_2 \mathbf{B}_u \, d\Omega$$

The first term corresponds to the contribution of the mechanical stiffness with $\mathbf{H}$ the Hooke tensor. As long as linear mechanics is considered, this first term is independent from the generalized displacements (the Hooke tensor and the integration domain are constant). The second term expresses the effect of electrostatic forces on effective stiffness. Conversely to the first term, the second one depends on the generalized

displacements through the integration domain, through $\mathbf{B}_u$ which is evaluated on the deformed configuration for the second integral and through matrix $\mathbf{F}_2$ whose definition is recalled later.

As in previous section, the integration domain depends on generalized displacements, and a variable substitution is used to transform the integral over the deformed domain $\Omega(\mathbf{q})$ into an integral over the reference domain Ω_{ref}. Consequently, the derivative of $\mathbf{K}_{uu}$ is:

$$\frac{\partial \mathbf{K}_{uu}}{\partial \mathbf{q}} = -\frac{1}{2}\int_{\Omega_{ref}} \left(\frac{\partial \mathbf{B}_u^T}{\partial \mathbf{q}} \mathbf{F}_2 \mathbf{B}_u \|\mathbf{J}\| + \mathbf{B}_u^T \frac{\partial \mathbf{F}_2}{\partial \mathbf{q}} \mathbf{B}_u \|\mathbf{J}\| + \mathbf{B}_u^T \mathbf{F}_2 \frac{\partial \mathbf{B}_u}{\partial \mathbf{q}} \|\mathbf{J}\| + \mathbf{B}_u^T \mathbf{F}_2 \mathbf{B}_u \frac{\partial \|\mathbf{J}\|}{\partial \mathbf{q}} \right) d\Omega \tag{A.6}$$

The matrix $\mathbf{B}_u$ is very similar to $\mathbf{B}_\phi$. Therefore, the interested reader may refer to section A.2.1 for the computational details of $\partial \mathbf{B}_u / \partial \mathbf{q}$. Moreover, the derivative of the Jacobian matrix has already been developed in section A.2.2.

The last unknown element in Eq. (A.6) is $\partial \mathbf{F}_2 / \partial \mathbf{q}$. $\mathbf{F}_2$ is defined by Rochus [91] as:

$$\mathbf{F}_2 = \begin{bmatrix} 2\varepsilon_{11}(E_{\xi_1})^2 & 0 & \cdots \\ 0 & 2\varepsilon_{22}(E_{\xi_1})^2 & \cdots \\ 2\varepsilon_{11}E_{\xi_1}E_{\xi_2} & \left(\varepsilon_{11}(E_{\xi_1})^2 + \varepsilon_{22}(E_{\xi_2})^2\right) & \cdots \\ -\left(\varepsilon_{11}(E_{\xi_1})^2 + \varepsilon_{22}(E_{\xi_2})^2\right) & 2\varepsilon_{22}E_{\xi_1}E_{\xi_2} & \cdots \\ \cdots & 2\varepsilon_{11}E_{\xi_1}E_{\xi_2} & -\left(\varepsilon_{11}(E_{\xi_1})^2 + \varepsilon_{22}(E_{\xi_2})^2\right) \\ \cdots & \left(\varepsilon_{11}(E_{\xi_1})^2 + \varepsilon_{22}(E_{\xi_2})^2\right) & 2\varepsilon_{22}E_{\xi_1}E_{\xi_2} \\ \cdots & 2\varepsilon_{11}(E_{\xi_2})^2 & 0 \\ \cdots & 0 & 2\varepsilon_{22}(E_{\xi_2})^2 \end{bmatrix}$$

Matrix $\mathbf{F}_2$ involves the two components of the electric field $E_{\xi 1}$ and $E_{\xi 2}$. The electric field vector is the gradient of the electric potential field. It is computed with the help of $\mathbf{B}_\phi$ matrix:

$$\mathbf{E} = \mathbf{B}_\phi \boldsymbol{\phi}$$

Where $\boldsymbol{\phi}$ is the vector of electric potential at each node of the element. Of course, the electric field depends on the domain deformation. Its derivative with respect to generalized displacements is:

$$\frac{\partial \mathbf{E}}{\partial \mathbf{q}} = \frac{\partial \mathbf{B}_\phi}{\partial \mathbf{q}} \boldsymbol{\phi} + \mathbf{B}_\phi \frac{\partial \boldsymbol{\phi}}{\partial \mathbf{q}}$$

The derivative of $\mathbf{B}_\phi$ has been computed previously in section A.2.1. $\boldsymbol{\phi}$ is composed of elements of $\mathbf{q}$ and therefore its derivative with respect to $\mathbf{q}$ results in a matrix composed of 0 and 1 according to the same rule as in Eq. (A.4).

In the scope of section 5.4.3, if one wants to compute $\frac{\partial \mathbf{K}_t}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l}$, then we have:

$$\frac{\partial \mathbf{E}}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l} = \frac{\partial \mathbf{B}_\phi}{\partial \mathbf{q}} \frac{d\mathbf{q}}{d\mu_l} \boldsymbol{\phi} + \mathbf{B}_\phi \boldsymbol{\phi}'$$

Where $\boldsymbol{\phi}'$ contains the elements of $d\mathbf{q}/d\mu_l$ corresponding to the DOFs included in $\boldsymbol{\phi}$.

A.4 Electromechanical coupling tangent stiffness $\mathbf{K}_{\phi u}$

As the tangent stiffness matrix is symmetric, the coupling matrix $\mathbf{K}_{\phi u}$ is equal to the transposed of the second coupling matrix $\mathbf{K}_{u\phi}$. Therefore, only the first one is considered in this appendix. The coupling matrix is defined by Rochus [91] as follows:

$$\mathbf{K}_{\phi u} = \int_{\Omega(\mathbf{q})} \mathbf{B}_\phi^T \mathbf{F}_1 \mathbf{B}_u \, d\Omega = \int_{\Omega_{ref}} \mathbf{B}_\phi^T \mathbf{F}_1 \mathbf{B}_u \, \|\mathbf{J}\| \, d\Omega$$

Once again, the integration domain depends on the generalized displacements. Consequently, a variable substitution is applied and the Jacobian determinant appears in the integral. All the elements of this definition have been presented earlier except $\mathbf{F}_1$ whose definition is:

$$\mathbf{F}_1 = \begin{bmatrix} \varepsilon_{11} E_{\xi_1} & \varepsilon_{22} E_{\xi_2} & \varepsilon_{11} E_{\xi_2} & -\varepsilon_{11} E_{\xi_1} \\ -\varepsilon_{22} E_{\xi_2} & \varepsilon_{22} E_{\xi_1} & \varepsilon_{11} E_{\xi_1} & \varepsilon_{22} E_{\xi_2} \end{bmatrix}$$

Matrix $\mathbf{F}_1$ involves the dielectric permittivity tensor $\boldsymbol{\varepsilon}$ and the electric field vector $\mathbf{E}$. As $\mathbf{F}_2$, $\mathbf{F}_1$ matrix depends on the generalized displacements through the electric field vector. Consequently, the derivative of $\mathbf{K}_{\phi u}$ is:

$$\frac{\partial \mathbf{K}_{\phi u}}{\partial \mathbf{q}} = \int_{\Omega_{ref}} \left(\frac{\partial \mathbf{B}_\phi^T}{\partial \mathbf{q}} \mathbf{F}_1 \mathbf{B}_u \, \|\mathbf{J}\| + \mathbf{B}_\phi^T \frac{\partial \mathbf{F}_1}{\partial \mathbf{q}} \mathbf{B}_u \, \|\mathbf{J}\| + \mathbf{B}_\phi^T \mathbf{F}_1 \frac{\partial \mathbf{B}_u}{\partial \mathbf{q}} \, \|\mathbf{J}\| + \mathbf{B}_\phi^T \mathbf{F}_1 \mathbf{B}_u \frac{\partial \, \|\mathbf{J}\|}{\partial \mathbf{q}} \right) d\Omega$$

As one can see, the derivatives on which the last equation relies have already been detailed previously. The derivative of $\mathbf{B}_\phi$ is treated in section A.2.1 and the derivative of $\mathbf{B}_u$ can be obtained by following the same reasoning. The derivative of the Jacobian matrix is described in section A.2.2. And finally, $\mathbf{F}_1$ can be derived like $\mathbf{F}_2$ whose definition is very similar (see section A.3).

A.5 Conclusion

Because it involves two physical fields, obtaining analytic sensitivities of the tangent stiffness matrix with respect to generalized displacements is more cumbersome than in the case of mechanical stiffness. However, the availability of analytic sensitivities is of a great interest since these sensitivities are more accurate and faster to evaluate than resorting to finite differences.

Bibliography

[1] M.M. Abdalla, C.K. Reddy, W.F. Faris, and Z. Gürdal. Optimal design of an electrostatically actuated microbeam for maximum pull-in voltage. *Computers & Structures*, 83(15–16):1320–1329, 2005.

[2] C. Acar and A. Shkel. *MEMS Vibratory Gyroscopes*. Springer, 2009.

[3] G. Allaire, F. Jouve, and A.-M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

[4] G. Allaire and S. M. Kaber. *Numerical Linear Algebra*. Springer, 2008.

[5] E.L. Allgower and K. Georg. Numerical path following. In P.G. Ciarlet and J.L. Lions, editors, *Techniques of Scientific Computing (Part 2)*, volume 5 of *Handbook of Numerical Analysis*, pages 3–207. Elsevier, 1997.

[6] E.L. Allgower and K. Georg. *Introduction to numerical continuation methods*. SIAM, Philadelphia, 2003.

[7] S. Amstutz, A.A. Novotny, and E.A. De Souza Neto. Topological derivative-based topology optimization of structures subject to Drucker-Prager stress constraints. *Computer Methods in Applied Mechanics and Engineering*, 233–236:123–136, 2012.

[8] A.A. Novotny an R.A Feijo, E. Taroco, and C. Padra. Topological sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering*, 192(7–8):803–829, 2003.

[9] F.D. Bannon, J.R. Clark, and C.T.-C. Nguyen. High frequency micromechanical filters. *IEEE Journal Solid-State Circuits*, 35(4):512–526, April 2000.

[10] K.-J. Bathe and E.N. Dvorkin. On the automatic solution of nonlinear finite element equations. *Computers & Structures*, 17(5–6):871–879, 1983.

[11] M.S. Bazaraa, H.D. Sherali, and C.M. Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, 2nd edition, 1993.

[12] A.D. Belegundu and S.D. Rajan. A shape optimization approach based on natural design variables and shape functions. *Computer Methods in Applied Mechanics and Engineering*, 66(1):87–106, 1988.

[13] D.J. Bell, T.J. Lu, N.A. Fleck, and S.M. Spearing. MEMS actuators and sensors: Observations on their performance and selection for purpose. *Journal of Micromechanics and Microengineering*, 15(7):S153–S164, 2005.

[14] A. Ben-Israel and T. Greville. *Generalized inverses theory and applications*. CMS Books in Mathematics. Springer New York, 2003.

[15] M.P. Bendsøe. Optimal shape design as a material distribution problem. *Structural Optimization*, 1:193–202, 1989.

[16] M.P. Bendsøe. Material interpolation schemes in topology optimization. *Archive of Applied Mechanics*, 69:635–654, 1999.

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

[18] M.P. Bendsøe, N. Olhoff, and J.E. Taylor. A variational formulation for multicriteria structural optimization. *Journal of Structural Mechanics*, 11(4):523–544, 1984.

[19] M.P. Bendsøe and O. Sigmund. *Topology Optimization: Theory, Methods, and Applications*. Springer Berlin / Heidelberg, 2003.

[20] L. Berke. An efficient approach to the minimum weight design of deflection limited structures. *USAF Technical Memorandum*, AFFDL-TM-70-4-FDTR, 1970.

[21] D.P. Bertsekas. *Nonlinear programming*. Athena Scientific, 2nd edition, 1999.

[22] B. Bourdin. Filters in topology optimization. *International Journal for Numerical Methods in Engineering*, 50:2143–2158, 2001.

[23] B. Brendel and E. Ramm. Linear and nonlinear stability analysis of cylindrical shells. *Computers & Structures*, 12(4):549–558, 1980.

[24] M. Bruggi and P. Duysinx. Topology optimization for minimum weight with compliance and stress constraints. *Structural & Multidisciplinary Optimization*, 46(3):369–384, 2012.

[25] T.E. Bruns and D.A. Tortorelli. Topology optimization of non-linear elastic structures and compliant mechanisms. *Computer Methods in Applied Mechanics and Engineering*, 190:3343–3459, 2001.

[26] M. Bruyneel, B. Colson, and A. Remouchamps. Discussion on some convergence problems in buckling optimisation. *Structural & Multidisciplinary Optimization*, 35(2):181–186, 2008.

[27] M. Bruyneel and P. Duysinx. Note on topology optimization of continuum structures including self-weight. *Structural & Multidisciplinary Optimization*, 29(4):245–256, April 2005.

[28] H.M. Chu and K. Hane. Design, fabrication and vacuum operation characteristics of two-dimensional comb-drive micro-scanner. *Sensors and Actuators A-Physical*, 165(2):422–430, 2011.

[29] R. Courant and D. Hilbert. *Methods of mathematical physics*, volume 1. Interscience Publishers, 1953.

[30] M.A. Crisfield. A fast incremental/iterative solution procedure that handles "snap-through". *Computers & Structures*, 13(1–3):55–62, 1981.

[31] M.U. Demirci and C. T.-C. Nguyen. Mechanically corner-coupled square microresonator array for reduced series motional resistance. *Journal of Microelectromechanical Systems*, 15(6):1419–1436, December 2006.

[32] P. Di Barba, A. Savini, and S. Slawomir. *Field Models in Electricity and Magnetism*. Springer, 2008.

[33] A. Diaz and O. Sigmund. Checkerboard patterns in layout optimization. *Structural & Multidisciplinary Optimization*, 10(1):40–45, August 1995.

[34] E.H. Dowell, R. Clark, D. Cox, H. Curtiss, J. Edwards, K. Hall, D. Peters, R. Scanlan, E. Simiu, F. Sisto, and T. Strganac. *A Modern Course in Aeroelasticity*, volume 116 of *Solid mechanics and its applications*. Kluwer Academic, 2004.

[35] P. Duysinx and M.P. Bendsøe. Topology optimization of continuum structures with local stress constraints. *International Journal for Numerical Methods in Engineering*, 43:1453–1478, 1998.

[36] P. Duysinx, W.H. Zhang, and C. Fleury. Sensitivity analysis with unstructured free mesh generators in 2-D and 3-D shape optimization. In J. Herskovits, editor, *Structural Optimization 93, The World Congress on Optimal Design of Structural Systems*, pages 205–212, August 1993.

[37] P. Duysinx, W.H. Zhang, H.G. Zhong, P. Beckers, and C. Fleury. Structural shape optimization with error control. *Proceedings of 1994 ASME, Design Technical Conference, Minneapolis, MN 11 - 14*, pages 51–59, September 1994.

[38] H.A. Eschenauer and N. Olhoff. Topology optimization of continuum structures: A review. *Applied Mechanics Review*, 54(4):331–390, July 2001.

[39] C. Fleury. *Le dimensionement automatique des structures élastiques*. PhD thesis, Université de Liège, 1978.

[40] C. Fleury. A unified approach to structural weight minimization. *Computer Methods in Applied Mechanics and Engineering*, 20(1):17–38, 1979.

[41] C. Fleury. CONLIN: an efficient dual optimizer based on convex approximation concepts. *Structural Optimization*, 1(2):81–89, June 1989.

[42] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009.

[43] J. K. Guest, J. H. Prvost, and T. Belytschko. 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, 2004.

[44] J.C. Halpin and S.W. Tsai. Effects of environmental factors on composite materials. *AFML-TR*, 67(423), June 1969.

[45] Z. Hashin and S. Shtrikman. A variational approach to the theory of effective magnetic permeability of multiphase materials. *Journal of Applied Physics*, 33:3125–3131, October 1962.

[46] Z. Hashin and S. Shtrikman. A variational approach to the theory of the elastic behaviour of multiphase materials. *Journal of the Mechanics and Physics of Solids*, 11:127–140, 1963.

[47] C. Jog and R. Haber. Stability of finite element models for distributed-parameter optimization and topology design. *Computer Methods in Applied Mechanics and Engineering*, 130(3–4):203–226, 1996.

[48] R. Jones. *Buckling of bars, plates, and shells*. Bull Ridge Publishing, 2006.

[49] H. Kardestuncer, D.H. Norrie, and F. Brezzi. *Finite element handbook*. McGraw-Hill reference books of interest: Handbooks. McGraw-Hill, 1987.

[50] I. Klapka, A. Cardona, and M. Géradin. An object-oriented implementation of the finite element method for coupled problems. *Revue Européenne des Elements Finis*, 7(5):469–504, 1998.

[51] Y.-Ch. Ko, J.-W. Cho, Y.-K. Mun, H.-G. Jeong, W.K. Choi, J.-W. Kim, Y.-H. Park, J.-B. Yoo, and J.-H. Lee. Eye-type scanning mirror with dual vertical combs for laser display. *Sensors and Actuators A-Physical*, 126(1):218 – 226, 2006.

[52] A. Konrad and M. Graovac. The finite element modeling of conductors and floating potentials. *IEEE Transactions on Magnetics*, 32(5):4329–4331, September 1996.

[53] L.A. Krog and N. Olhoff. Optimum topology and reinforcement design of disk and plate structures with multiple stiffness and eigenfrequency objectives. *Computers & Structures*, 72(4–5):535–563, August 1999.

[54] H.N. Kwon, J.-H Lee., K. Takahashi, and H. Toshiyoshi. Microxy stages with spider-leg actuators for two-dimensional optical scanning. *Sensors and Actuators A-Physical*, 130–131(SPEC. ISS.):468–477, 2006.

[55] T.S. Kwon, B.C. Lee, and W.J. Lee. An approximation technique for design sensitivity analysis of the critical load in non-linear structures. *International Journal for Numerical Methods in Engineering*, 45(12):1727–1736, 1999.

[56] M. Langelaar, G.H. Yoon, Y.Y. Kim, and F. Van Keulen. Topology optimization of planar shape memory alloy thermal actuators using element connectivity parameterization. *International Journal for Numerical Methods in Engineering*, 88(9):817–840, 2011.

[57] B. S. Lazarov and O. Sigmund. Filters in topology optimization based on helmholtz-type differential equations. *International Journal for Numerical Methods in Engineering*, 86(6):765–781, 2011.

[58] S.S. Li, Y.W. Lin, Z. Ren, and C.T.-C. Nguyen. Disk-array design for suppression of unwanted modes in micromechanical composite-array filters. In *19th IEEE Int. Conf. on Micro Electro Mechanical Systems (MEMS'06)*, pages 866–869, January 2006.

[59] E. Lindgaard and E. Lund. Nonlinear bucking optimization of composite structures. *Computer Methods in Applied Mechanics and Engineering*, 199(37–40):2319–2330, 2010.

[60] E. Lindgaard and E. Lund. Optimization formulations for the maximum nonlinear buckling load of composite structures. *Structural & Multidisciplinary Optimization*, 43(5):631–646, 2010.

[61] E. Lindgaard and E. Lund. A unified approach to nonlinear buckling optimization of composite structures. *Computers & Structures*, 89(3–4):357–370, 2011.

[62] E. Lindgaard, E. Lund, and K. Rasmussen. Nonlinear buckling optimization of composite structures considering "worst" shape imperfections. *International Journal of Solids and Structures*, 47(22–23):3186–3202, 2010.

[63] Z. Liu. Some methods to solve singularity in electrostatics with FEM. Technical report, Institute for Microsystem Technology, Albet-Ludwigs University Freigburg, 2002.

[64] Z. Liu, J.G. Korvink, and M. Reed. Solving singularities in electrostatics with high-order FEM. In *Sensors, 2003. Proceedings of IEEE*, volume 2, pages 872–876, 2003.

[65] E. Lund. Buckling topology optimization of laminated multi-material composite shell structures. *Composite Structures*, 91(2):158–167, 2009.

[66] Z. Luo, N. Zhang, J. Ji, and T. Wu. A meshfree level-set method for topological shape optimization of compliant multiphysics actuators. *Computer Methods in Applied Mechanics and Engineering*, 223–224:133–152, 2012.

[67] Z.-D. Ma, N. Kikuchi, and H.-C. Cheng. Topological design for vibrating structures. *Computer Methods in Applied Mechanics and Engineering*, 121(1–4):259–280, 1995.

[68] M. Madou. *Fundamentals of microfabrication: the science of miniaturization.* CRC Press, 2nd edition, 2002.

[69] M.D. Mankame and G.K. Ananthasuresh. Topology synthesis of electrothermal compliant mechanisms using line elements. *Structural & Multidisciplinary Optimization*, 26:209–218, 2004.

[70] K. Maute and D.M. Frangopol. Reliability-based design of MEMS mechanisms by topology optimization. *Computers & Structures*, 81(8–11):813–824, 2003.

[71] Y. Mochida, M. Tamura, and K. Ohwada. A micromachined vibrating rate gyroscope with independent beams for the drive and detection modes. *Sensors and Actuators A-Physical*, 80:170–178, March 2000.

[72] E. Munoz. *Global optimization for structural design by generalized benders' decomposition.* PhD thesis, Technical University of Denmark, 2010.

[73] M.M. Neves, H. Rodrigues, and J.M. Guedes. Generalized topology design of structures with a buckling load criterion. *Structural Optimization*, 10(2):71–78, 1995.

[74] J. Nocedal and S.J. Wright. *Numerical Optimization.* Springer series in operations research and financial engineering. Springer, 2nd edition, 2006.

[75] N. Olhoff, M.P. Bendsøe, and J. Rasmussen. On cad-integrated structural topology and design optimization. *Computer Methods in Applied Mechanics and Engineering*, 89, 1992.

[76] Oofelie. Object Oriented Finite Elements Led by Interactive Executor. Internet resources: www.open-engineering.com.

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

[78] S. Pamidighantam, R. Puers, K. Baert, and H.A.C. Tilmans. Pull-in voltage analysis of electrostatically actuated beam structures with fixed-fixed and fixed-free end conditions. *Journal of Micromechanics and Microengineering*, 12(4):458–464, 2002.

[79] S. Pamidighantam, R. Puers, and H.Tilmans. Pull-in voltage analysis of fixed-fixed beams. In *MME2001*, pages 269–272, 2001.

[80] J. S. Park and K. K. Choi. Design sensitivity analysis of critical load factor for nonlinear structural systems. *Computers & Structures*, 36(5):823–838, 1990.

[81] B.N. Parlett. *The symmetric eigenvalue problem*. Prentice-Hall, 1980.

[82] C.B.W. Pedersen, T. Buhl, and O. Sigmund. Topology synthesis of large displacement compliant mechanisms. *International Journal for Numerical Methods in Engineering*, 50(12):2683–2705, 2001.

[83] N. Pedersen. Maximization of eigenvalues using topology optimization. *Structural & Multidisciplinary Optimization*, 20(1):2–11, August 2000.

[84] X. Qian and O. Sigmund. Topological design of electromechanical actuators with robustness toward over- and under-etching. *Computer Methods in Applied Mechanics and Engineering*, 253:237 – 251, 2013.

[85] S.A. Ragon, Z. Gürdal, and L.T. Watson. A comparison of three algorithms for tracing nonlinear equilibrium paths of structural systems. *International Journal of Solids and Structures*, 39:689–698, February 2002.

[86] M. Raulli and K. Maute. Topology optimization of electrostatically actuated microsystems. *Structural & Multidisciplinary Optimization*, 30(5):342–359, November 2005.

[87] J.R. Reitz, F.J. Milford, and R.W. Christy. *Foundations of electromagnetic theory*. Addison-Wesley, 3rd edition, 1979.

[88] J.-F. Remacle, N. Chevaugeon, E. Marchandise, and C. Geuzaine. Efficient visualization of high-order finite elements. *International Journal for Numerical Methods in Engineering*, 69(4):750–771, 2007.

[89] R.J. Roark and W.C. Young. *Formulas for Stress and Strain*. McGraw-Hill, 1989.

[90] V. Rochus. *Finite element modelling of strong electro-mechanical coupling in MEMS*. PhD thesis, University of Liège, 2006.

[91] V. Rochus, D.J. Rixen, and J.-C. Golinval. Monolithic modelling of electro-mechanical coupling in micro-structures. *International Journal for Numerical Methods in Engineering*, 65(4):461–493, 2006.

[92] K. Schittkowski. NLPQL: A Fortran subroutine solving constrained nonlinear programming problems. *Annals of Operation research*, 5(2):485–500, June 1986.

[93] L.A. Schmit. Structural design by systematic synthesis. *Proceedings of the* 2^{nd} *ASCE Conference on Electronic Computation*, pages 105–132, 1960.

[94] S.D. Senturia. *Microsystem Design.* Kluwer Academic Publisher, 2001.

[95] J.A. Sethian and A. Wiegmann. Structural boundary design via level set and immersed interface methods. *Journal of Computational Physics*, 163(2):489–528, 2000.

[96] A.P. Seyranian, E. Lund, and N. Olhof. Multiple eigenvalues in structural optimization problems. *Structural Optimization*, 8:207–227, 1994.

[97] A.P. Seyranian and O.G. Privalova. The Lagrange problem on an optimal column: old and new results. *Structural Optimization*, 25:393–410, October 2003.

[98] O. Sigmund. On the design of compliant mechanisms using topology optimization. *Mechanics of Structures and Machines*, 25(4):493–526, 1997.

[99] O. Sigmund. Design of multiphysic actuators using topology optimization - Part I: One material structures. - Part II: Two-material structures. *Computer Methods in Applied Mechanics and Engineering*, 190(49–50):6577–6627, 2001.

[100] O. Sigmund. Morphology-based black and white filters for topology optimization. *Structural & Multidisciplinary Optimization*, 33:401–424, January 2007.

[101] O. Sigmund. Manufacturing tolerant topology optimization. *Acta Mechanica Sinica*, 25(2), 2009.

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

[103] A. Sihvola. *Electromagnetic Mixing formulas and Applications.* Institution of Electrical Engineers, 1999.

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

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

[106] I.C. Taig and R.I. Kerr. Optimization of aircraft structures with multiple stiffness requirements. *AGARD Second Symposium on Structural Optimization*, AGARD CP 123: paper 16, 1973.

[107] K. Takahashi, H.N. Kwon, K. Saruta, M. Mita, H. Fujita, and H. Toshiyoshi. A two-dimensional f-θ micro optical lens scanner with electrostatic comb-drive xy-stage. *IEICE Electronics Express*, 2(21):542–547, 2005.

[108] K. Takahashi, M. Mita, H. Fujita, and H. Toshiyoshi. A high fill-factor comb-driven xy-stage with topological layer switch architecture. *IEICE Electronics Express*, 3(9):197–202, 2006.

[109] N.P. Van Dijk, M. Langelaar, and F. Van Keulen. Explicit level-set-based topology optimization using an exact heaviside function and consistent sensitivity analysis. *International Journal for Numerical Methods in Engineering*, 91(1):67–97, 2012.

[110] P.F. Van Kessel, L.J. Hornbeck, R.E. Meier, and M.R. Douglass. A MEMS-based projection display. *Proceedings of the IEEE*, 86(8):1687–1704, 1998.

[111] L. Van Miegroet. *Generalized shape optimization using XFEM and level set description*. PhD thesis, University of Liège, 2012.

[112] L. Van Miegroet and P. Duysinx. Stress concentration minimization of 2d filets using x-fem and level set description. *Structural & Multidisciplinary Optimization*, 33(4–5):425–438, 2007.

[113] W. van Spengen, R. Puers, and I. de Wolf. A physical model to predict stictions in MEMS. *Journal of Micromechanics and Microengineering*, 190, 2001.

[114] V.K. Varadan, K.J. Vinoy, and K.A. Jose. *RF MEMS and Their Applications*. John Wiley & Sons, October 2002.

[115] K. Wang and C.T.-C. Nguyen. High-order medium frequency micromechanical electronic filters. *Journal of Microelectromechanical Systems*, 8(4), December 1999.

[116] M.Y. Wang and S. Wang. Bilateral filtering for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 63:1911–1938, April 2005.

[117] M.Y. Wang, X. Wang, and D. Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1–2):227–246, 2003.

[118] B. Wardle, P.A. Lagace, and M.A. Tudela. Buckling response of transversely loaded composite shells, part 2: Numerical analysis. *AIAA Journal*, 42(7):1465–1473, July 2004.

[119] J.E. Warren. *Nonlinear stability analysis of frame-type structures with random geometric imperfections using a total-lagrangian finite element formulation*. PhD thesis, Virginia Polytechnic Institute and State University, January 1997.

[120] L.T. Watson, M. Sosonkina, R.C. Melville, A.P. Morgan, and H.F. Walker. Algorithm 777: HOMPACK90: A suite of Fortran 90 codes for globally convergent homotopy algorithms. *ACM Trans. Math. Softw.*, 23:514–549, December 1997.

[121] C. Wu and J. Arora. Design sensitivity analysis of non-linear buckling load. *Computational Mechanics*, 3(2):129–140, 1988.

[122] M. Wu and W. Fang. Design and fabrication of mems devices using the integration of mumps, trench-refilled molding, drie and bulk silicon etching processes. *Journal of Micromechanics and Microengineering*, 15(3):535–542, 2005.

[123] Q. Xia, T. Shi, and M. Wang. A level set based shape and topology optimization method for maximizing the simple or repeated first eigenvalue of structure vibration. *Structural & Multidisciplinary Optimization*, 43:473–485, 2011.

[124] S. Xu, Y. Cai, and G. Cheng. Volume preserving nonlinear density filter based on heaviside functions. *Structural & Multidisciplinary Optimization*, 41(4):495–505, 2010.

[125] J.J. Yao. RF MEMS from a device perspective. *Journal of Micromechanics and Microengineering*, 10:9–38, April 2000.

[126] Z.J. Yao, S. Chen, S. Eshelman, D. Denniston, and C. Goldsmith. Micromachined low-loss microwave switches. *Journal of Microelectromechanical Systems*, 8(2):129–134, June 1999.

[127] N. Yazdi, F. Ayazi, and K. Najafi. Micromachined inertial sensors. *Proceedings of the IEEE*, 86(8):1640–1659, August 1998.

[128] L. Yin and G.K. Ananthasuresh. A novel topology design scheme for the multiphysics problems of electro-thermally actuated compliant micromechanisms. *Sensors and Actuators*, 97–98:599–609, April 2002.

[129] G.H. Yoon and O. Sigmund. A monolithic approach for topology optimization of electrostatically actuated devices. *Computer Methods in Applied Mechanics and Engineering*, 194:4062–4075, 2008.

[130] X.M. Zhang, F.S. Chau, C. Quan, Y.L. Lam, and A.Q. Liu. A study of the static characteristics of a torsional micromirror. *Sensors and Actuators A-Physical*, 90(1–2):73–81, 2001.

[131] M. Zhou, R. Fleury, Y.K. Shyy, H. Thomas, and J.M. Brennan. Progress in topology optimization with manufacturing constraints. In *9th AIAA/ISSMO Symposium on Mutlidisciplinary Analysis and Optimization*, Atlanta, Georgia, September 2002. AIAA.

[132] M. Zhou, Y.K. Shyy, and H.L. Thomas. Checkerboard and minimum member size control in topology optimization. *Structural & Multidisciplinary Optimization*, 21:152–158, March 2001.

[133] M. Zhou, Y.K. Shyy, and H.L. Thomas. Topology optimization with manufacturing constraints. In *4th World Congress of Structural and Multidisciplinary Optimization, Dalian, China*, June 2001.

[134] O.C. Zienkiewicz and J.S. Campbell. Shape optimization and sequential linear programming. In R.H. Gallagher and O.C. Zienkiewicz, editors, *Optimum Structural Design*, pages 109–126. Wiley, New York, 1973.

[135] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method (5th edition) Volume 1: The Basis*. Butterworth-Heinemann, Oxford, 2000.

[136] O.C. Zienkiewicz and R.L. Taylor. *The Finite Element Method (5th edition) Volume 2: Solid Mechanics*. Butterworth-Heinemann, Oxford, 2000.