POLITECNICO DI MILANO
Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea in Ingegneria Matematica

A NUMERICAL MODEL FOR THE PREDICTION OF DAMPING IN MEMS RESONATORS WITH EXTENDED FREQUENCY RANGE

Supervisors: Prof. Giacomo LANGFELDER
Prof. Attilio FRANGI

Co-supervisor: Ing. Patrick FEDELI

Master Thesis of:
Arianna CHIESA
Matr. 852319

Academic Year 2017-2018

Contents

1 Introduction **1**

1.1 Introduction to MEMS 1

1.2 Microfabrication of MEMS devices 2

1.3 The Spring-Mass-Damper Model 3

1.3.1 Torsional Spring-Mass-Damper 8

1.4 The damping coefficient b 9

1.4.1 Thermo-elastic damping (TED) 10

1.4.2 Anchor losses 11

1.4.3 Fluid damping 11

1.4.4 The Test Particle Monte Carlo method 19

1.5 Conclusions 20

2 Numerical model: part 1 **22**

2.1 Numerical model for free-molecule flows 22

2.2 Simplifying assumptions 26

2.3 Numerical implementation 30

2.3.1 Testing the visibility condition 31

2.4 Bugs in the previous implementation 34

2.4.1 A false-positive in the identification of visible quadrangles 34

2.4.2 Change of status for triangles 35

2.5 Failure in identifying some invisible quadrangles 36

2.6 Structures geometry 37

3 Numerical model: part 2 **40**

3.1 Improvements on the code: fixing of the bugs 40

3.1.1 False-positives in the identification of visible quadrangles 40
3.1.2 Change of status for triangles 41
3.2 Improvements on the code: overcoming of the imprecisions . . . 43
3.2.1 Failure in identifying some invisible quadrangles 43
3.3 Performance of the new implementation 46
3.3.1 Visibility performance 46
3.3.2 Time performance 48

4 Experimental validation 53
4.1 Test structures 53
4.2 Numerical results on damping estimation 56
4.3 Description of the instrumentation 58
4.4 First campaign of measurements: probe station 60
4.4.1 Step approach 62
4.4.2 Pulse-train approach 62
4.4.3 Short-burst signal approach 63
4.4.4 First campaign results 64
4.5 Second campaign of measurements: socket 66
4.5.1 Measurement repeatability 67
4.5.2 Second campaign results 68
4.6 Summary 69

List of Figures

1.1 The ThELMA micromachinig process by STMicroelectronics. (1) First thermal oxidation. (2) Deposition and modellization of interconnections. (3) Second thermal oxidation. (4) Epitaxial growth of the structural layer. (5) Trench etch. (6) Etching of the sacrificial layer and metal deposition. 4
1.2 Rapresentation of the absolute and the relative frames with the point-like mass P. 6
1.3 The Spring-Mass-Damper system 8
1.4 Thermo-elastic dissipation due to temperature gradient in a spring. 10
1.5 Slide film damping between a fixed substrate and a moving plate. 14
1.6 Squeeze film damping between a moving plate and the substrate. 16
1.7 Types of collisions between molecules and a plate. 17
1.8 Squeeze film damping in a holed mass. 19
1.9 Example of a typical geometry of a MEMS gyroscope. 20

2.1 A moving wall accelerates molecules increasing the number of collisions. 25
2.2 Classification of surfaces according to their visibility. 31
2.3 Levels of the visibility test: Levels 1 and 2 detect invisible surfaces, while Level 3 looks for visible surfaces. 33
2.4 (a): a wide surface B completely covers surfaces A and C. (b) Example of configuration where surfaces A and C are wrongly detected as “totally visible” in Levels 3 though B partially covers them. 35

2.5 Example of geometry where some triangles wrongly changed their status to "visible", when a triangle goes from "source triangle" to "tested triangle" and viceversa. 36
2.6 Top view of an example of geometry where an invisible surface (D) is not recognized by tests in Levels 1 and 2. 37
2.7 Structures geometry used as inputs of the code. 39

3.1 Excerpt of the *hiddenelems* routine, where the main parameter, which discriminates if a test triangle is behind or on the same plane of the source triangle, is highlighted. 42
3.2 Excerpt of the code of the splitting routine: the calls to the subroutines *split* and *els_partition* are highlighted in yellow. . . 45
3.3 Steps of the surfaces visibility. For each step, the source surface or the source triangle is highlighted in white. 47
3.4 Comparisons between time performance of the old code with bugs (red) and the new and improved one (blue). 51

4.1 Optical top view of the 30-μm-thick test structures fabricated to validate the proposed damping modelling. Rows represent varying frequency, columns represent different vertical gaps. . . 54
4.2 Predicted damping force on elementary holed blocks as a function of frequency and gap. 57
4.3 Picture of the setup used for measurements in the clean room. . 59
4.4 Scheme of the functioning of the MCP. 60
4.5 Summary of the three adopted measurements approaches. 61
4.6 Experimental results using the probe station. Red $\rightarrow$ 10 kHz, yellow $\rightarrow$ 20 kHz, blue $\rightarrow$ 50 kHz and cyan $\rightarrow$ 100 kHz. Besides, in each color group: the highest curve with markers $\rightarrow$ 1.8 μm gap; the lowest curve with markers $\rightarrow$ 1.2 μm gap. 64
4.7 Graphical explanation of the effect causing differences in measurements depending on the mounting. 66
4.8 Ceramic carrier where the die is mounted. It is also connected to the pads by means of wire bondings. 66

4.9 Results of the repeatability measurements on a sample with 1.8 μm vertical gap and 100 kHz resonant frequency. 67
4.10 Comparison between the results of the first campaign (cyan markers) and the second campaign (blue markers). The dashed lines are the average measured Q, while the red line is the theoretical value. 68
4.11 Experimental results of Q factor measurements (circles) compared to low-frequency model (squares) and the new proposed model (diamonds), as a function of frequency. 70

List of Tables

3.1 Computational time taken by each block of the old code, for each of the five considered geometries. 49
3.2 Comutational time taken by each block of the new code, for each of the five considered geometry. 50

4.1 Pull-in voltages of each type of structure. 56
4.2 Identifier codes of the eight devices in a package. 62

Sommario

Il lavoro presentato in questa tesi si focalizza sul miglioramento di un nuovo modello numerico, utile per la stima del coefficiente di damping in MEMS inerziali. Questo modello supera quelli già esistenti in letteratura, poiché risulta essere più accurato, computazionalmente più efficiente e di vasta applicazione. È da far notare che il nuovo modello è stato formulato a partire da un modello precedente valido, però, solo per basse frequenze di risonanza, mentre il modello su cui si è lavorato vale anche per le alte frequenze e può quindi essere applicato a un esteso range di frequenze.

In questo lavoro, sono stati inizialmente corretti gli errori trovati nella precedente implementazione, che causano una sottostima della dissipazione. Il modello perciò è più accurato rispetto al vecchio. In un secondo momento, la parte di codice inerente alla visibilità delle strutture, viene migliorata implementando routine aggiuntive che aiutano nel risparmiare tempo computazionale e nel velocizzare il calcolo complessivo delle forze dissipative che agiscono sulle superfici del MEMS. Dunque il modello presentato risulta essere anche più efficiente dal punto di vista computazionale, soprattutto se comparato con quelli presenti in letteratura che fanno uso del metodo Test Particle Monte Carlo che è un algoritmo lento per natura.

Successivamente viene validato sperimentalmente conducendo due campagne di misure nel Laboratorio di MEMS e microsensori del Politecnico di Milano, su otto categorie diverse di dispositivi forniti da STMicroelectronics caratterizzati da differenti frequenze di risonanza e differente distanza verticale tra MEMS e substrato.

Confronti tra i risultati sperimentali, ottenuti con le due campagne di cui sopra, e i risultati numerici, forniti sia dal vecchio che dal nuovo modello, permettono di validare quest'ultimo.

Abstract

The present work focuses on the improvement of a new numerical model for the prediction of gas damping in inertial MEMS. This model overcomes those existing in literature, since it results to be more accurate, more computationally efficient and of wide application. Note that it has been formulated starting from an existing one, which is valid only for low resonance frequencies, while the model under study holds even at higher frequencies and can be applied to an extended frequency range.

In this work, errors from the previous implementation, which lead to the underestimation of damping, are firstly corrected. It is therefore more accurate with respect to the old model. In a second moment, the visibility part of the code is improved by implementing additional routines, which help in saving computational time and speeding up the whole computation of damping forces acting on MEMS surfaces. Therefore the presented model results also to be more computationally efficient, especially if compared to those existing in literature which make use of the Test Particle Monte Carlo method, that is inherently a slow algorithm.

Then it is experimentally validated, by conducting two measurements campaigns in the Laboratory of MEMS and microsensors of Politecnico di Milano, on eight types of devices provided by STMicroelectronics and characterized by different resonance frequencies and different vertical gap.

Comparisons between experimental results, obtained from the two campaigns above, and numerical results, provided both by the old model and by the new one, lead to the validation of the latter.

Chapter 1

Introduction

1.1 Introduction to MEMS

The recent developments in smart electronics are characterized by MEMS: "MEMS" is the acronym standing for Micro Electro Mechanical Systems. As the name implies, they are devices composed by electrical and mechanical parts with characteristic dimension in the order of micrometers (μm). Thanks to the combination of electrical and mechanical parts, MEMS can easily interact with the surroundings: this allows the investigation of physical phenomena such as linear motion, rotation, propagation of acoustic waves, ecc. Indeed *sensors* collect information from the environment measuring such phenomena. Then the electronics processes signals from sensors and guide the *actuators* in order to react and control the external environment to achieve a specific goal.

Thus MEMS can be exploited in a large number of situations and nowadays consumer and automotive are the main sectors for their applications.

Among MEMS sensors, those employed in almost every area are inertial MEMS: *accelerometers* and *gyroscopes*.

In general inertial sensors exploit inertial forces acting on an object to determine its dynamic behaviour. External forces acting on a body cause an acceleration (resulting in the displacement of the body) and/or a change of its orientation (and thus of its angular position).

The same working principle applies for inertial MEMS. Their mechanical

part includes a seismic mass, which is attached to a fixed substrate by means of anchor points and suspended through springs that are partially free to move under the action of external forces. The electrical part is instead composed by electrodes to apply and readout signals.

Then, if an external force moves the suspended frame, its displacement can be detected by the electrodes and it is thus possible to quantify the force acting on the device.

1.2 Microfabrication of MEMS devices

Beside the fact that MEMS sensors can be exploited in various fields, as mentioned in the previous section, they have a unitary low-cost due to the consolidated CMOS-like fabrication technique. This represents a great advantage since a unitary low cost allows a large-scale production for the same type of device.

This is indeed the paradigm which the success of microelectronics is based on: the fabrication of a large number of components on wafer, each one identical to the others. Therefore the production techniques of these components should be extremely precise, in order to fabricate micro structures, and repeatable, in order to produce a huge number of copies of the same type of device.

This innovative technique used for the fabrication of 3D and 2D structures on the micrometer scale is called *micromachining* and it was introduced by Peterson in 1982. The structural element which micromachining is based on is a silicon wafer, usually characterized by a diameter of $25 - 300mm$, where layers of material are deposited or etched. Indeed the creation of moving parts within planar structures is inevitably linked to removal of materials around the target object and this explains the need of etching.

Different micromachining techniques have been developed by MEMS companies, but each of these processes is in general characterized by the same steps: deposition and etching of material.

Since the devices studied in this work are provided by STMicroelectronics, which uses the ThELMA (Thick Epitaxial Layer for Micro-gyroscopes and

Accelerometers) surface micromachining process to produce MEMS, a detailed explanation of this technique is given in the following [1]. As suggested by the acronym, the ThELMA technique combines variably thick and thin polysilicon layers for structures and interconnections, enabling the integration of accelerometer and gyroscope mechanical elements in a single chip. It consists of the following steps:

1. *First thermal oxidation of the substrate*: it occurs at high temperatures (1100 °C) and in an O_2 environment; it is operated on a Silicon substrate to create a first sacrificial SiO_2 layer.

2. *Depositation of interconnections*: a thin layer of polysilicon is deposited; it is then selectively etched to create paths, which will later form buried electrical interconnections to bring signals to the mechanical parts.

3. *Second thermal oxidation*: it increases the sacrificial layer thickness, which is selectively etched to give access to electrical interconnections.

4. *Epitaxial growth of the structural layer*: polysilicon has been grown on top of the sacrificial layer reaching a $25 - 30\ \mu m$ thickness.

5. *Trench etch*: the structural layer is deeply etched until the oxide layer is reached; by means of this process suspended and anchored parts are created.

6. *Etching of the sacrificial layer and metal deposition*: it leads to the release of the suspended structures. Then the metal contact (tipically alluminium or gold) is deposited to define the access pads. In the end all the structure is encapsulated by a cap to protect it from the environment.

See Fig.1.1 to better understand each step of the ThELMA micromachining process.

1.3 The Spring-Mass-Damper Model

Since in most MEMS devices there is the need to measure displacements and rotations induced by external forces, it is necessary to fix the formalism to

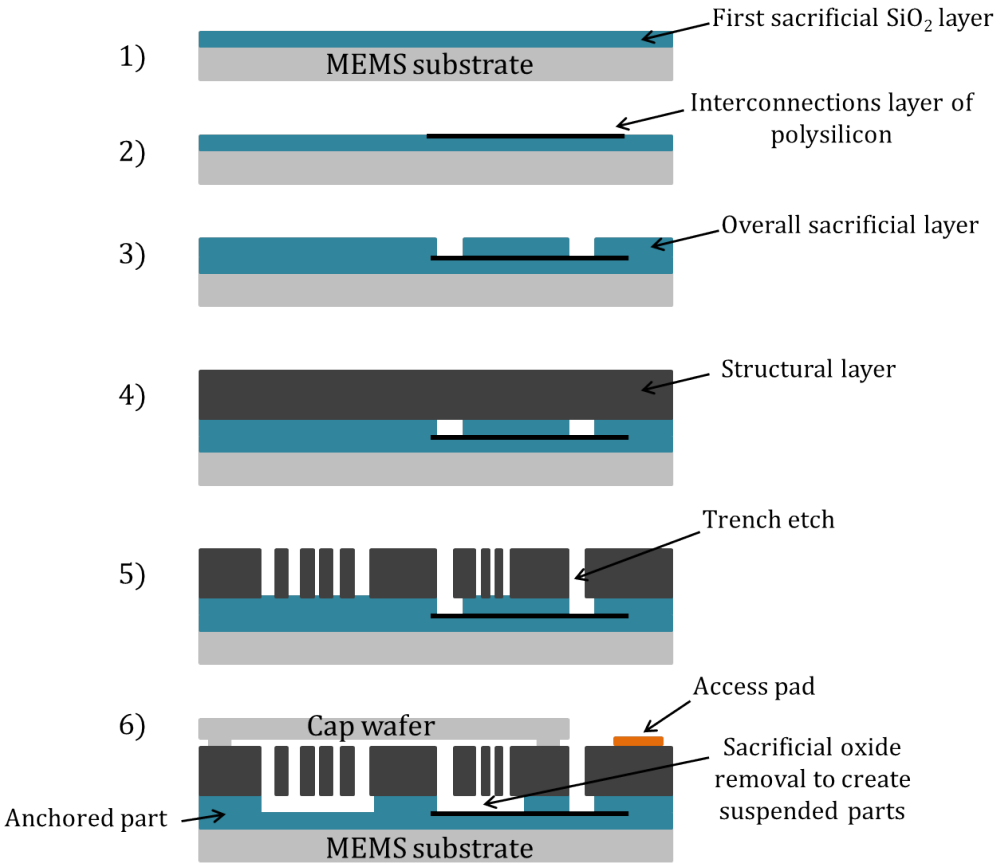


Figure 1.1: The ThELMA micromachinig process by STMicroelectronics. (1) First thermal oxidation. (2) Depositation and modellization of interconnections. (3) Second thermal oxidation. (4) Epitaxial growth of the structural layer. (5) Trench etch. (6) Etching of the sacrificial layer and metal deposition.

describe the working principle of MEMS. Thus in the following it is provided a model which is at the basis of the solution of any problem related to MEMS: the *spring-mass-damper model*.

First of all it is useful to give a definition of an *inertial reference system*: it is a frame where the first Newton's law applies.

> "In an inertial frame of reference, an object either remains at rest or continues to move at a constant velocity, unless acted upon by a force".

All inertial frames are coherent one to the others, in the sense that they are in a state of constant, rectilinear motion one another. Physical laws take

the same form in all inertial frames.

When dealing with *non-inertial reference frames* instead, the laws of physics depend upon the particular frame of reference and the usual physical forces must be supplemented by fictitious forces.

Thus in practical situations Earth can be considered as the inertial (absolute) reference frame; the package where the device is mounted in (such as the mobile phone, the car...) as the non-inertial (relative) reference frame; while the suspended mass, which is partially free to move relative to the non-inertial frame, can be considered as the sensing element reference frame and is approximated with a point-like mass.

The two reference frames and the point-like mass newly introduced can be schematized by the following quantities (see Fig.1.2):

- O_{xyz}: inertial (absolute) reference frame;
- $O'_{x'y'z'}$: non-inertial (relative) reference frame;
- P: point-like mass;
- $\mathbf{r}_{O'_a}$: vector describing the position of O' with respect to the absolute reference;
- $\mathbf{r}_{P_r}$: vector describing the posistion of point P with respect to the relative reference;
- $\mathbf{r}_{P_a}$: vector describing the position of point P with respect to the absolute reference;
- Ω_a: vector describing the angular velocity of the relative reference frame;
- $\mathbf{v}_{O'_a}$, $\mathbf{a}_{O'_a}$, $\mathbf{v}_{P_r}$, $\mathbf{a}_{P_r}$, $\mathbf{v}_{P_a}$ and $\mathbf{a}_{P_a}$: vectors of the corresponding velocities and accelerations.

Now the complete expressions for position, velocity and acceleration of P

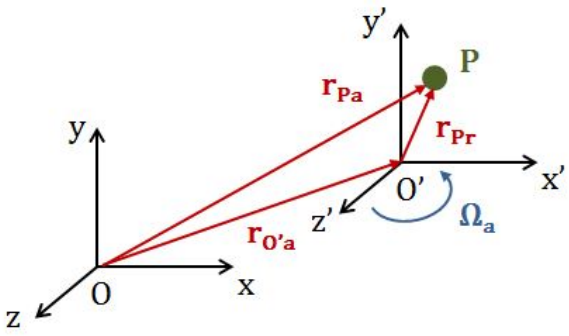


Figure 1.2: Rapresentation of the absolute and the relative frames with the point-like mass P.

with respect to the absolute reference frame are given by:

$$\mathbf{r}_{P_a} = \mathbf{r}_{P_r} + \mathbf{r}_{O'_a} \ ,$$
$$\mathbf{v}_{P_a} = \mathbf{v}_{P_r} + \mathbf{v}_{O'_a} + (\Omega_a \times \mathbf{r}_{P_r}) \ ,$$
$$\mathbf{a}_{P_a} = \mathbf{a}_{P_r} + \mathbf{a}_{O'_a} + (\dot{\Omega}_a \times \mathbf{r}_{P_r}) + \Omega_a \times (\Omega_a \times \mathbf{r}_{P_r}) + 2(\Omega_a \times \mathbf{v}_{P_r}) \ ,$$

where $(\Omega_a \times \mathbf{r}_{P_r})$ is the tangential velocity, $\Omega_a \times (\Omega_a \times \mathbf{r}_{P_r})$ is the angular acceleration, which together with $\mathbf{a}_{O'_a} + (\dot{\Omega}_a \times \mathbf{r}_{P_r})$ is the dragging acceleration, and $\mathbf{a}_{P_r}$ is the initial acceleration. In this general formulation effects of fictitious forces have been taken into account too, indeed $2(\Omega_a \times \mathbf{v}_{P_r})$ is the Coriolis acceleration.

In order to derive the equation for the spring-mass-damper system, it is necessary to calculate the motion of the non-inertial system with respect to the absolute one. To do so, the motion of P relative to O', described through fictitious forces, is exploited. For the sake of initial simplicity the rotation and the associated velocity, acceleration and Coriolis terms can be neglected ($\Omega_a = 0$). Thus O' is in conditions of translational motion only with respect to O and the acceleration of the non-inertial system with respect to the absolute frame results to be: $\mathbf{a}_{O'_a} = \mathbf{a}_{P_a} - \mathbf{a}_{P_r}$.
In every inertial reference system a force acting on the mass P can be evaluated

as:

$$F_{P_a} = m\mathbf{a}_{P_a} \ , \tag{1.1}$$

while in the non-inertial frame it becomes:

$$F_{P_r} = m\mathbf{a}_{P_r} = m\mathbf{a}_{P_a} - m\mathbf{a}_{O'_a} = F_{P_a} + F_{\text{inertial}} \ , \tag{1.2}$$

where F_{P_a} is a real Newton force and F_{inertial} is a fictitious force due to the relative motion of O'.

Since the suspended mass P is confined in small dimensions sensor package, its maximum allowed displacement is in the order of $10\mu m$. Thus it is more realistic to imagine it attached to the non-inertial frame by means of a spring of stiffness k [N/m]. Therefore as the mass displaces only by a small amount from the frame, it is subject to a reactive elastic force:

$$F_{\text{el}} = -kx_{P_r} \ .$$

Besides a damper is also introduced in the system configuration in order to make it as similar as possible to real situations. A damper generates a reactive force proportional to the mass velocity through the damping coefficient b [N/m $\cdot$ s]:

$$F_{\text{b}} = -b\dot{x}_{P_r} = -b\frac{dx_{P_r}}{dt} \ .$$

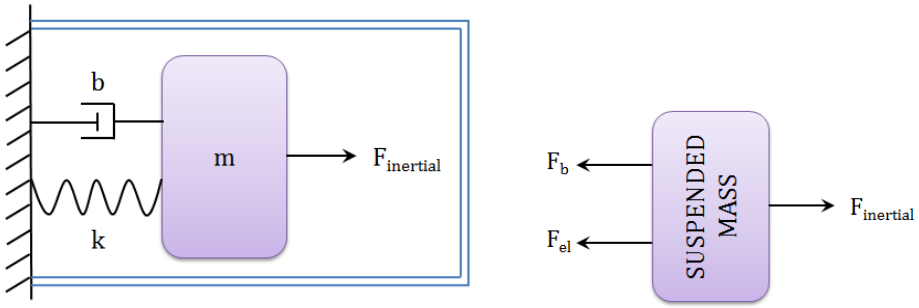
See Fig.1.3 for a representation of the spring-mass-damper system and a schematic drawing of forces acting on the suspended mass.

Considering now forces acting on the suspended mass, one ends up with the following equality in the absolute reference:

$$F_{P_a} = m\frac{d^2x_{P_a}}{dt} = -kx_{P_r} - b\frac{dx_{P_r}}{dt} \ . \tag{1.3}$$

Thanks to eq.(1.2) it becomes:

$$m\frac{d^2x_{O'_a}}{dt} + m\frac{d^2x_{P_r}}{dt} = -kx_{P_r} - b\frac{dx_{P_r}}{dt} \ , \tag{1.4}$$



(a) Representation of the spring-mass-damper system

(b) Scheme of the forces acting on the suspended mass

Figure 1.3: The Spring-Mass-Damper system

thus

$$m\frac{d^2x_{P_r}}{dt} + b\frac{dx_{P_r}}{dt} + kx_{P_r} = -m\frac{d^2x_{O'_a}}{dt} = -m\mathbf{a}_{O'_a} = F_{\text{inertial}} \ . \tag{1.5}$$

In conclusion the general equation for the spring-mass-damper system is given by:

$$m\ddot{x} + b\dot{x} + kx = F_{\text{inertial}} = F_{\text{ext}} \ . \tag{1.6}$$

The whole discussion can be extended to the torsional motion by taking into account the Coriolis force.

1.3.1 Torsional Spring-Mass-Damper

It is valuable to notice that some MEMS operate through torsions of structural elements, rather than through deflections. Therefore the torsional spring-mass-damper system is presented in the following. The already shown linear spring-mass-damper model still holds, but now one has to deal with different variables than in the previous case: an angular displacement θ [rad] instead of a linear one x, the moment of inertia I [kg $\cdot$ m^2] instead of the mass of the device m and the torque M [N $\cdot$ m] instead of the force F acting on the mass.

Then the previous model becomes:

$$I\ddot{\theta} + b_\theta\dot{\theta} + k_\theta\theta = M_{\text{ext}} \ . \tag{1.7}$$

Note that the stiffness k_θ [N · m] and damping coefficient b_θ [kg/s · m^2] become their torsional counterparts. Therefore, as a consequence, their units of measurement are different.

1.4 The damping coefficient b

In order to evaluate MEMS performances, a relevant parameter is the *quality factor* Q, given by $Q = \dfrac{\omega_0 m}{b}$ $\left(Q = \dfrac{\omega_0 I}{b_\theta} \text{ in case of a torsional device}\right)$. It is inversely proportional to the damping coefficient b, which provides the rate of energy dissipation in MEMS. Thus it is important and useful (and also quite challenging) to give a closed form expression of b.

The analysis of the damping coefficient may be complicated due to different interacting geometries in MEMS structure (such as springs, frames, plates...) and different sources of energy dissipation. Thus depending on different dissipation mechanisms, it results that the main types of damping are: fluid damping, thermoelastic damping (TED) and damping due to anchor losses (propagation and consequent dissipation of elastic energy outside the device). A detailed description of these three will be given in the following.

The focus now is in the understanding of the physical origin of the quality factor and thus of b.

In resonators, Q is defined as the energy stored in the system divided by the energy dissipated per radiant of vibration cycle:

$$Q = 2\pi \cdot \frac{U_{\text{stored in the system}}}{U_{\text{dissipated per cycle}}} \ . \tag{1.8}$$

Thus in order to have a low energy loss, the quality factor should be as high as possible [2].

To understand which phenomena yield to low Q, it has to be considered that energy dissipation is the combination of intrinsic energy losses U_{int} within

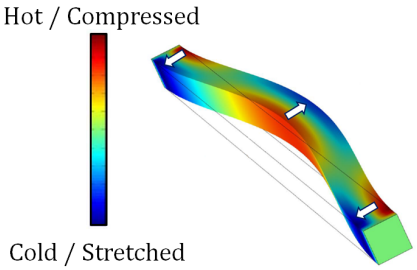


Figure 1.4: Thermo-elastic dissipation due to temperature gradient in a spring.

the structure (TED and anchor losses) and losses due to the interaction of the device surfaces with the surrounding fluid U_s:

$$Q = 2\pi \cdot \frac{U_{\text{stored in the system}}}{U_{\text{int}} + U_s} = \frac{1}{1/Q_{\text{int}} + 1/Q_s} \,. \tag{1.9}$$

Note that losses due to surface phenomena become more and more relevant in comparison with volumetric effects, as the device shrinks, because the volume-to-surface ratio scales down with linear dimension. Thus in MEMS devices the impact of surface effects increases accordingly. This is why in this work the focus has been kept on fluid damping. Nonetheless a description of those mechanisms responsible for intrinsic energy losses U_{int} (thermoelastic damping and anchor losses) is provided.

1.4.1 Thermo-elastic damping (TED)

Now that all the initial observations have been made, it can be explained what TED consists of (make reference to Fig.1.4).

Energy losses can occur in vibrating structures when an internal temperature gradient arises. Indeed when a solid body, for example a sustaining spring, is deformed, then a strain gradient is produced (higher temperature for compressed parts and lower for stretched ones) leading to a temperature gradient. This induced temperature gradient generates an irreversible heat flow through the vibrating structure with a consequent loss of energy.

This phenomenon was first observed by Zener in his work [3] for the rect-

angular cross-sectional beam-type geometry:

$$Q_{TED} = \left(\frac{f_M^2 + f_T^2}{f_M f_T} \right) \cdot \frac{C_p \rho}{\alpha^2 T E} , \tag{1.10}$$

where f_M and f_T are the mechanical and the thermal resonance frequencies respectively, C_p is the specific heat capacity, α is the thermal expansion coefficient, ρ is the solid density, while T and E are the absolute temperature and the Young's modulus. Thus the energy loss due to TED is characterized by the relative magnitude of two frequencies: if these frequencies are close, then the damping is maximum as the heat exchange is facilitated by the combination of two resonant phenomena. If they are distant, then b is minimized.

1.4.2 Anchor losses

The other phenomenon which causes intrinsic energy dissipation in MEMS devices is represented by anchor losses. Indeed in a MEMS resonator, while the resonator structure vibrates, the harmonic load excites elastic waves, which can propagate through the elements used to anchor the resonator to the substrate. Thus the elastic wave that propagates into the supporting media dissipates part of the vibration energy. It is desirable that the anchors are nodes on the waves propagation. If so, they would have null energy and thus there would be no waves transmission to the substrate. For more details see [2].

1.4.3 Fluid damping

As already mentioned above, besides intrinsic energy dissipation by TED and anchor losses, energy can be lost also by interactions of the device surfaces with the surrounding fluids. This phenomenon is called fluid damping.

Its modelling is more complicated than the previous damping mechanisms because of:

- different interacting fluid-structure geometries generated by the elements (suspended or fixed) constituting the MEMS device, such as springs, frames, plates, electrodes;

- different types of surface-to-fluid interactions: slide-film damping and squeeze-film damping;
- different nature of the fluid according to the Knudsen number, as discussed below.

The Knudsen number

In order to give a detailed description of slide-film damping and squeeze-film damping, it is necessary to distinguish between different natures of the fluid flow. Thus it is worthwhile to introduce the Knudsen number.

Denoting by L one characteristic length of the device (or of the portion of the device of interest for a partial damping calculation) and by λ the mean free path of the molecules of the gas in the considered {T,p} operating conditions, the Knudsen number is defined as:

$$Kn = \frac{\lambda}{L} \ . \tag{1.11}$$

Different Knudsen numbers may exist for one device, depending on which fluid regime is modelled:

- if $\lambda << L \longrightarrow Kn << 1$: each molecule of the fluid interacts mainly with other molecules and not with the device walls. The fluid can be thus considered and treated as continuum, therefore the molecular nature of the fluid can be neglected. This regime is well described by the complete Navier-Stokes equation.
- if $\lambda >> L \longrightarrow Kn >> 1$: the fluid can be considered in its molecular structure, thus only collisions of gas particles with device walls are taken into account, since particles have a very low probability to interact one another. This regime is described by the Boltzmann equation for particles.

Transition zones exist between continuum and free molecular flow. They are characterized by an intermediate status where features of continuum and free molecular behaviour are simultaneously present.

To select the correct flow model when describing a portion of MEMS a structure, four regions have been highlighted according to the corresponding Knudsen numbers:

- $Kn \leq 0.01$: continuum flow with no-slip boundary condition (that is no relative movement between the surface and the adjacent fluid at the boundary);
- $0.01 \leq Kn \leq 0.1$: continuum flow with slip boundary condition;
- $0.1 \leq Kn \leq 1$: transitional flow;
- $Kn > 1$: free molecular flow.

Note that the mean free path λ is inversely proportional to the pressure. Therefore, being P_0 the atmospheric pressure ($P_0 = 1.01325$ bar) and λ_0 the mean free path of molecules in air under standard conditions, which is about $\lambda_0 = 65$ μm, λ can be expressed as [4]:

$$\lambda = \lambda_0 \frac{P_0}{P} \ . \tag{1.12}$$

Since MEMS resonators used in this work operate under low-pressure conditions, about $P = 0.7$ mbar in package, λ results to be 92.6 μm. Moreover these devices (as later discussed in Chapter 4) are distinguished into two groups according to their vertical gap between the suspended holed mass and the substrate: a first group of test devices is characterized by a vertical gap $g_1 = 1.2$ μm, while the second groups has a gap $g_2 = 1.8$ μm. Thus their Knudsen number can be evaluated:

$$Kn_1 = \frac{\lambda}{g_1} = 77.16 \tag{1.13}$$

$$Kn_2 = \frac{\lambda}{g_2} = 51.44 \ . \tag{1.14}$$

For both classes of devices the Knudsen number results to be greater than 1 indicating the regime of free molecular flow.

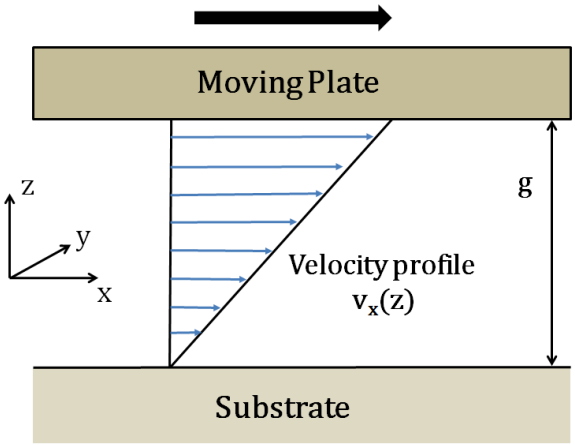


Figure 1.5: Slide film damping between a fixed substrate and a moving plate.

Slide-film damping

One of the two main types of fluid damping in inertial MEMS is slide-film damping. It occurs when two separated surfaces slide parallel to each other. The flow around an object follows the object's movement due to normal (pressure) and tangential (shear) forces acting on the fluid.

The damping coefficient in this case can be found as [4]:

$$b_{\text{slide}} = \frac{A}{g}\mu \ , \tag{1.15}$$

where A is the overlapping area between the two separated surfaces, g is the vertical gap between them and μ is the viscosity coefficient of the surrounding gas.

For low values of the Knudsen number (thus in the continuum flow regime), the fluid closed to a fixed part tends to remain still due to the absence of slip boundary conditions (see Fig.(1.5)).

With the Knudsen number growing larger, thus for larger mean free paths λ, a slip between the fixed wall and the adjacent fluid layer appears. In other words, the flow velocity $v_x(z)$ of a moving fluid layer is not zero at the boundary with a stationary wall, while the flow velocity of the boundary layer is proportional to the velocity gradient in the $z-$direction and the mean free path.

As the Knudsen number grows larger, a switchover from the transitional regime to the free molecular regime occurs. In this last case, slip means that particles approaching the fixed wall have different velocities from particles leaving it.

The slip of a boundary layer can be accounted for by introducing a *reduced effective viscosity* μ_{eff} given by:

$$\mu_{\text{eff}} = \frac{\mu}{1 + f(Kn)} \ , \tag{1.16}$$

where $f(Kn)$ is a function of the Knudsen number which does not have a determined expression. However for the slide-film damping case, an empirical approximation of $f(Kn)$ is given by Veijola *et al.* in [5]:

$$\mu_{\text{eff}} = \frac{\mu}{1 + 2Kn + 0.2Kn^{0.788}e^{-Kn/10}} \ . \tag{1.17}$$

Note that if $Kn \to 0$, then $f(Kn) \to 0$ and therefore $\mu_{\text{eff}} \to \mu$. Thus the effective viscosity is a generalization of the viscosity μ. Indeed, since there are accepted and functioning models for damping in the continuum flow regime (see equation 1.15), the idea behind the effective viscosity is to extend these existing models towards higher Knudsen numbers in order to give an estimate for the damping coefficient also in the transitional and the free molecular regimes.

Thus a relation for the slide damping coefficient, which holds in almost every regime, is given by:

$$b_{\text{slide}} = \frac{A}{g}\mu_{\text{eff}} \ , \tag{1.18}$$

where the viscosity μ has been replaced by its effective counterpart μ_{eff} [6].

However this is not a general model: it holds only for finite surfaces which are a particular case of "ideal" geometries (infinite surfaces). Besides, for high levels of gas rarefaction the concept of viscosity becomes meaningless, as explained in details in the following.

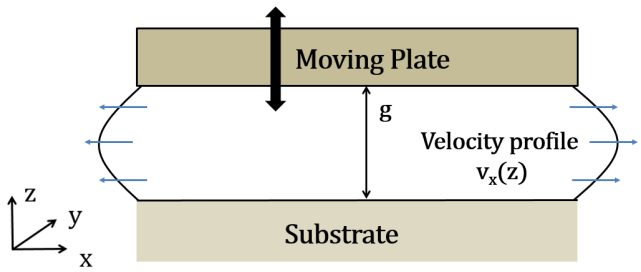


Figure 1.6: Squeeze film damping between a moving plate and the substrate.

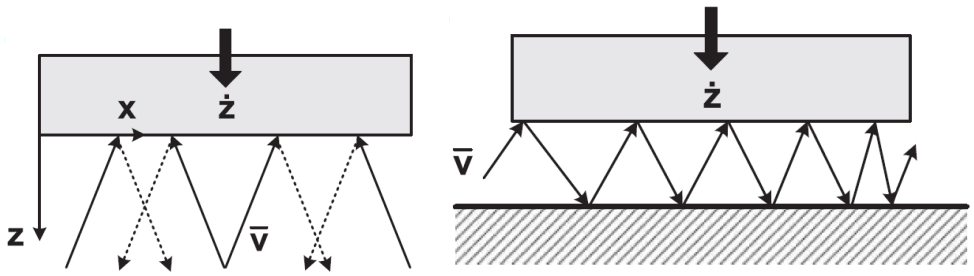
Squeeze-Film Damping

Squeeze-film damping is a second type of fluid damping. It occurs when two parallel surfaces move towards or away from each other and squeeze or expand the fluid film in between. If two plates move towards each other the pressure increases and the gas is squeezed out of the gap, otherwise if two plates move away from each other, then the pressure decreases and gas is drawn into the gap. See Fig.1.6.

In this case the same argument of the previous paragraph holds: in the equations for squeeze-film damping in rarefied gas, the coefficient of viscosity μ should be replaced by an effective one μ_{eff}, which is dependent on the pressure via Knudsen number. As in the previous case, Veijola *et al.* in [5] give an empirical approximation of it:

$$\mu_{\text{eff}} = \frac{\mu}{1 + 9.638 Kn^{1.159}} \ . \tag{1.19}$$

However in [7] Bao and Yang argue that for a pressure much lower than an atmospheric pressure, the collisions among the gas molecules are so reduced that the gas can hardly be considered as a viscous fluid. Indeed if $P \ll P_0$, then $Kn \gg 1$ (by equations (1.11) and (1.12)) and the continuum flow model is no more valid. As a consequence, the concept of *viscosity* would become questionable. For this reason they propose a new model for squeeze damping in free molecular flow regime. Their work starts from that of Christian presented in [8]: the interaction between gas molecules is neglected and the damping force on an oscillating plate is found by the momentum transfer rate from the



(a) Head on collisions of an oscillating plate with molecules.

(b) Multiple collisions with a molecule entering the gap between a moving plate and a fixed substrate.

Figure 1.7: Types of collisions between molecules and a plate.

vibrating plate to the surrounding gas through the collisions between the plate and the molecules.

Consider the gas damping force acting on a plate of area A oscillating in its normal direction (z-direction) as shown in Fig.1.7(a). It experiences head-on collisions with molecules of mass m_g. The velocity component responsible for the momentum transfer of molecules to the plate is v_z. Collisions on the front side of the plate cause a momentum transfer of $\Delta P_+ = 2m_g(v_z + \dot{z})$ and on the back side of $\Delta P_- = 2m_g(v_z - \dot{z})$, where $\dot{z}$ is the velocity of the moving plate.

If the number of molecules in a unit volume is N, then the fraction of them with velocities in the range of v_z to $v_z + dv_z$ is dN_{v_z}. Therefore the number of molecular collisions on a unit area and in unit time is

$$(v_z + \dot{z})dN_{v_z} = N(v_z + \dot{z})f(v_z)dv_z \ , \tag{1.20}$$

where $f(v_z)$ is the Maxwellian distribution:

$$f(v_z) = \sqrt{\frac{m_g}{2\pi kT}}\exp\left(- \frac{m_g v_z^2}{2kT}\right) . \tag{1.21}$$

Thus the pressure caused by the collisions on the front plate surface is

$$P_+ = 2m_g N \int_{-\dot{z}}^{\infty} (v_z + \dot{z})^2 f(v_z)\mathrm{d}v_z \ , \tag{1.22}$$

while on the back is

$$P_{-} = 2m_g N \int_{\dot{z}}^{\infty} (v_z - \dot{z})^2 f(v_z) \mathrm{d}v_z \ . \tag{1.23}$$

The net acting pressure can be easily calculated as $(P_{+} - P_{-})$ and results to be:

$$P = 4\sqrt{\frac{2}{\pi}}\sqrt{\frac{M}{RT}} p_a \dot{z} \ , \tag{1.24}$$

where M is the molar mass, R the universal gas constant and p_a the ambient pressure. The damping coefficient is therefore:

$$b = A\frac{P}{\dot{z}} = 4p_a A\sqrt{\frac{2}{\pi}}\sqrt{\frac{M}{RT}} \ . \tag{1.25}$$

However this is a general model for gas damping of an isolated object and not for squeeze-film damping. Thus to overcome this problem, Bao and Yang extend Christian's model proposing the Energy Transfer model.

They introduced energy loss of the plate caused by molecules entering the gap between the moving plate and the substrate with some lateral velocity component and colliding multiple times with the plate, as shown in Fig.1.7(b). The extra energy gained by a molecule travelling through the gap is given by:

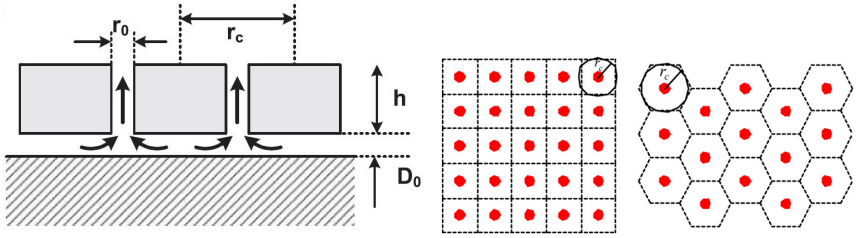
$$\Delta E = \frac{1}{2} m_g \left[\frac{2w v_{z0}^2}{(D_0 - z) v_{xy0}} \dot{z} + \frac{w^2 v_{z0}^2}{(D_0 - z)^2 v_{xy0}^2} \dot{z}^2 \right] \ , \tag{1.26}$$

where v_{z0} and v_{xy0} are the velocity components of the molecule entering in the z-plane and in-plane respectively; D_0 is the maximum allowed oscillation amplitude; w is the width of the considered plate.

Thus the damping coefficient for squeeze-film damping in free molecular flow regime is:

$$b_{\text{squeeze}} = (2\pi)^{\frac{2}{3}} \frac{w L^2 p_a}{D_0} \sqrt{\frac{M}{RT}} \ . \tag{1.27}$$

It results to be greater than that of Christian's by a factor $L/(16\pi D_0)$. This perfectly makes sense since Chiristian's model would underestimate the damping if used to calculate it in the squeeze case, because it concerns only



(a) Flow in a perforated plate (b) Holes arranged in periodic lattices

Figure 1.8: Squeeze film damping in a holed mass.

an isolated object and not two facing surfaces. Bao and Yang's model instead correct its estimate.

Note that in presence of a holed mass a lowering in the damping coefficient occurs. Indeed if the plate is moving down, fluid underneath is pushed into the holes and creates a pipe flow as illustrated in Fig.1.8(a). If the plate is very thin, then pressure at the hole's entry is equal to the ambient pressure and as a result there is no pressure difference. This means that boundary conditions periodically change over the plate.

Sometimes etching holes are implemented purposely to reduce squeeze-film damping and usually they are arranged in periodic lattice structures as shown in Fig.1.8(b) [4].

1.4.4 The Test Particle Monte Carlo method

Other approaches, different from the one presented in this work, make use of the Test Particle Monte Carlo (TPMC) method. This probabilistic numerical method has been exploited to predict the damping coefficient in free-molecular flows in [9]. A large number of typical molecular trajectories is computed and these collectively predict the behaviour of the real system. Since intermolecular collisions can be neglected in free-molecular regime, the trajectories are independent one another. However this is a statistical approach and therefore a statistical noise is introduced, which affects the precision on damping prediction. In order to filter it out long analysis are required, slowing down the

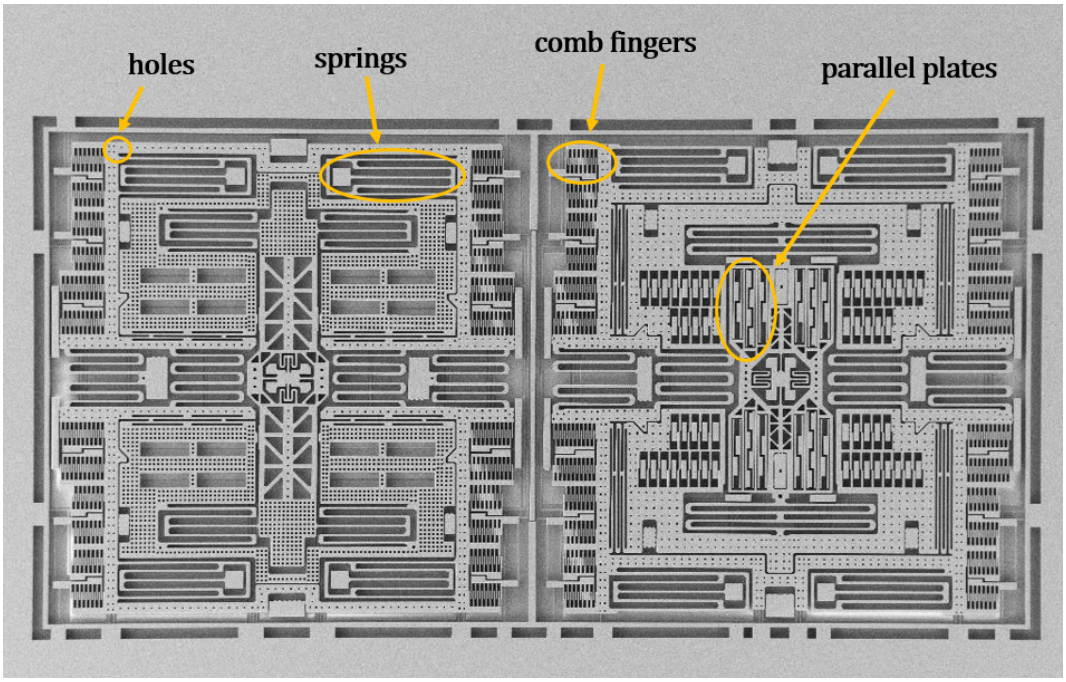


Figure 1.9: Example of a typical geometry of a MEMS gyroscope.

calculation.

1.5 Conclusions

Summarizing what has been presented so far, models based on the modified viscosity [5] result to be inaccurate when a gas is highly rarefied: the concept of viscosity becomes meaningless and thus the effective viscosity. On the other hand the different proposed formulations [7] provide satisfying results only in very specific conditions of rarefaction of gas and with somewhat ideal geometries, far from the typical geometries of MEMS. An example of a MEMS gyroscope is depicted in Fig.1.9: one can see that its geometry is composed by different structures (parallel plates, comb finger, springs and holes) which interact one another and therefore complicate the modelling.

On the contrary, an approach which makes use of the TPMC method can be exploited with more complicated and realistic geometries of MEMS and in every condition of gas rarefaction. However it is a very time consuming

approach which introduces a statistical noise.

Therefore there is the need to propose a new method which holds for every condition of rarefaction of a gas, contrary to effective viscosity methods, and results to be more computationally efficient than TPMC method. Such model will be presented in the following chapter.

Chapter 2

Numerical model: theory and implementation

In this chapter the derivation of the numerical model formulated to overcome the limitations of the models presented in Chapter 1 is provided.

In order to evaluate mechanical dissipation in micro-systems working at low pressures, a boundary integral elements approach has been proposed (details can be found in [10]).

A BIE technique has been found to be a suitable approach for estimating MEMS dissipation since micromechanical structures are too geometrically complicated to analyze analytically, therefore a numerical approach is needed. Moreover, mechanical dissipation is mainly due to pressure and drag forces generated by the gas surrounding the mechanical structure. Thus only forces on the structure surfaces are taken into account, leading to surface-only integral equations.

2.1 Numerical model for free-molecule flows

According to the kinetic theory of gases ([11], [12], [13]), macroscopic quantities like density ρ and mean velocity $\boldsymbol{v}$ can be expressed as moments of the

distribution function $f(\boldsymbol{x}, \boldsymbol{\xi}, t)$ in the velocity space:

$$\rho(\boldsymbol{x}, t) = \int_{\mathbb{R}^3} f(\boldsymbol{x}, \boldsymbol{\xi}, t)\, \mathrm{d}\boldsymbol{\xi} \ , \tag{2.1}$$

$$\rho(\boldsymbol{x}, t)\boldsymbol{v}(\boldsymbol{x}, t) = \int_{\mathbb{R}^3} f(\boldsymbol{x}, \boldsymbol{\xi}, t)\boldsymbol{\xi}\, \mathrm{d}\boldsymbol{\xi} \ , \tag{2.2}$$

where $\boldsymbol{x}$ denotes the position and $\boldsymbol{\xi}$ denotes the molecular velocity.

In the free-molecule regime collisions among molecules are neglected, thus the total material derivative of f along the molecular trajectory vanishes:

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi} \cdot \nabla_x f = 0 \ , \tag{2.3}$$

which represents the Boltzmann equation with null right-hand side since intermolecular collisions are neglected.

In the applications envisaged herein, the micro-system is immersed in a gas domain extending to infinity where rest conditions are assumed and f is expressed by the Maxwellian distribution at the equilibrium:

$$f(\boldsymbol{x}, \boldsymbol{\xi}, t) = f_0(\boldsymbol{\xi}) = \frac{\rho_0}{(2\pi\mathcal{R}T_0)^{3/2}} \exp\Big(- \frac{|\boldsymbol{\xi}|^2}{2\mathcal{R}T_0} \Big) \ . \tag{2.4}$$

Note that temperature T_0 and density ρ_0 are far-field average conditions and $\mathcal{R}$ is the universal gas constant divided by the molar mass. Here the case where a package is present is only a specific case of this general situation.

Focusing on a specific position $\boldsymbol{x}$ on the boundary of the micro-system: its unit normal vector $\boldsymbol{n}(\boldsymbol{x}, t)$ is assumed to point into the fluid domain, while $\boldsymbol{w}(\boldsymbol{x}, t)$ is its velocity, thus it represents the wall velocity. With these assumptions the distribution function $f(\boldsymbol{x}, \boldsymbol{\xi}, t)$ for any $\boldsymbol{\xi}$ such that $(\boldsymbol{\xi} - \boldsymbol{w}(\boldsymbol{x}, t)) \cdot \boldsymbol{n}(\boldsymbol{x}, t) < 0$ corresponds to molecules impinging on the surface at time t. If molecules are coming from the far region, then $f(\boldsymbol{x}, \boldsymbol{\xi}, t) = f_0$. Otherwise they have been emitted from a specific position $\boldsymbol{y}$ on the boundary at time $t - r/\xi$, where r is the distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ and $\xi = |\boldsymbol{\xi}|$.

As a consequence of Eq.(2.3), one has:

$$f(\boldsymbol{x}, \boldsymbol{\xi}, t) = f(\boldsymbol{y}, \boldsymbol{\xi}, t) \ , \tag{2.5}$$

where

$$\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{r}, \quad \boldsymbol{r} = -r\frac{\boldsymbol{\xi}}{\xi} \tag{2.6}$$

for any couple $\boldsymbol{x}$, $\boldsymbol{y}$.

Each molecule impinging on the surface ($(\boldsymbol{\xi} - \boldsymbol{w})\cdot\boldsymbol{n} < 0$) interacts with it and is eventually re-emitted ($(\boldsymbol{\xi} - \boldsymbol{w}) \cdot \boldsymbol{n} > 0$) according to a specific distribution function which depends on the surface properties. Since silicon surfaces are originated from etching procedures (as described in Sec.1.2), they are very rough and at common temperatures microscopically rough surfaces are characterized by diffuse reflections of molecules: therefore the diffuse reflection model will be assumed in the following. This means that molecules are re-emitted from a given solid surface S_R according to the following distribution function:

$$f(\boldsymbol{x}, \boldsymbol{\xi}, t) = \frac{\rho_w(\boldsymbol{x}, t)}{(2\pi\mathcal{R}T_w(\boldsymbol{x}, t))^{3/2}}\exp\Big(- \frac{|\boldsymbol{\xi} - \boldsymbol{w}(\boldsymbol{x}, t)|^2}{2\mathcal{R}T_w(\boldsymbol{x}, t)}\Big) , \tag{2.7}$$

for $(\boldsymbol{\xi} - \boldsymbol{w})\cdot\boldsymbol{n} > 0$. It represents a half-space isotropic reflection of re-emitted molecules whose velocity is independent of incident velocity.

In the distribution function above T_w is the wall temperature, while $\boldsymbol{w}(\boldsymbol{x}, t)$ is the wall velocity.

The function ρ_w in (2.7) is proportional to the flux of incoming molecules:

$$\rho_w(\boldsymbol{x}, t) = \Big(\frac{2\pi}{\mathcal{R}T_0}\Big)^{1/2} \int_{\mathbb{R}^3,(\boldsymbol{\xi}-\boldsymbol{w})\cdot\boldsymbol{n}<0} |(\boldsymbol{\xi} - \boldsymbol{w}(\boldsymbol{x}, t))\cdot\boldsymbol{n}(\boldsymbol{x}, t)| f(\boldsymbol{x}, \boldsymbol{\xi}, t)\, \mathrm{d}\boldsymbol{\xi} \ . \tag{2.8}$$

This last relation represents a mass balance at every point of the surface and is the starting point of the integral formulation.

Now using Eq.(2.7) and denoting by $\boldsymbol{y}$ the point corresponding to $\boldsymbol{x}$ ac-

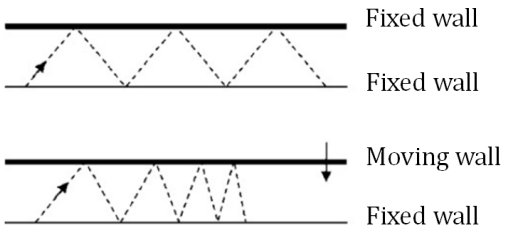


Figure 2.1: A moving wall accelerates molecules increasing the number of collisions.

cording to relation (2.6), the wall density results to be:

$$\begin{aligned}\rho_w(\boldsymbol{x},t) = &-\left(\frac{2\pi}{\mathcal{R}T_0}\right)^{1/2}\int_{\mathbb{R}^3_0}(\boldsymbol{\xi}-\boldsymbol{w}(\boldsymbol{x},t))\cdot\boldsymbol{n}(\boldsymbol{x},t)f_0(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi}\,+\\ &-\frac{1}{2\pi(\mathcal{R}T_0)^2}\int_{\mathbb{R}^3_S}(\boldsymbol{\xi}-\boldsymbol{w}(\boldsymbol{x},t))\cdot\boldsymbol{n}(\boldsymbol{x},t)\rho_w\Big(\boldsymbol{y},t-\frac{r}{\boldsymbol{\xi}}\Big)\times\\ &\times\exp\left(-\frac{\left|\boldsymbol{\xi}-\boldsymbol{w}\Big(\boldsymbol{y},t-\frac{r}{\xi}\Big)\right|^2}{2\mathcal{R}T_0}\right)\mathrm{d}\boldsymbol{\xi}\ .\end{aligned} \tag{2.9}$$

$\mathbb{R}^3_0$ denotes the space of molecular velocities such that $(\boldsymbol{\xi}-\boldsymbol{w})\cdot\boldsymbol{n}<0$, where $\boldsymbol{\xi}$ corresponds to molecules coming from the far field.
$\mathbb{R}^3_S$ instead denotes the space of molecular velocities such that $(\boldsymbol{\xi}-\boldsymbol{w})\cdot\boldsymbol{n}<0$, where $\boldsymbol{\xi}$ corresponds to molecules coming from other solid surfaces.

Note that the Maxwellian distribution is centered at $\boldsymbol{w}$ and that implies:

- if $\boldsymbol{w}\cdot\boldsymbol{n}>0$: molecules re-emitted by the surface will be in the average faster than those impinging on the surface.
- if $\boldsymbol{w}\cdot\boldsymbol{n}<0$: molecules re-emitted will be in the average slower than those impinging on the surface.

If a moving wall approaches a fixed one the consequence is that molecules entering the gap get accelerated and the number of collisions increases, as depicted in Fig.2.1.

This seems to complicate the calculation of energy dissipation, however some simplifying assumptions can be assumed according to MEMS properties leading to the integral formulation.

2.2 Simplifying assumptions

Small perturbation condition

For a large class of MEMS the velocity of shuttle surfaces $\boldsymbol{w}$ is generally small with respect to the average thermal molecular speed, hence $|\tilde{\boldsymbol{w}}| = \dfrac{|\boldsymbol{w}|}{\sqrt{2\mathcal{R}T}} \ll 1$. Since a perturbation of the equilibrium of gas is small, the problem can be linearized.

Defining a new variable $\boldsymbol{u} = \boldsymbol{\xi} - \boldsymbol{w}$, the first term in equation (2.9) is linearized as:

$$\begin{aligned}&\left(\frac{2\pi}{RT_0}\right)^{1/2} \int_{\mathbb{R}^3_0} (\boldsymbol{\xi} - \boldsymbol{w}(\boldsymbol{x},t)) \cdot \boldsymbol{n}(\boldsymbol{x},t) f_0(\boldsymbol{\xi})\, \mathrm{d}\boldsymbol{\xi} \\ &\simeq \frac{2\rho_0}{\pi} \int_{\mathbb{R}^3_0} \tilde{u}_n \exp(-\tilde{u}^2)(1 - 2\tilde{u}_n \tilde{w}_n(\boldsymbol{x},t) - 2\tilde{\boldsymbol{u}}_t \cdot \tilde{\boldsymbol{w}}_t(\boldsymbol{x},t))\, \mathrm{d}\tilde{\boldsymbol{u}}\ ,\end{aligned} \tag{2.10}$$

where quantities with tilde are normalized with respect to $\sqrt{2\mathcal{R}T_0}$; n and t subscripts indicate respectively the normal and the tangential component of a vector.

Thus by means of the definition of Maxwellian distribution f_0 and of the hypothesis of small perturbation $|\tilde{\boldsymbol{w}}| \ll 1$, f and ρ_w are linearized as:

$$f(\boldsymbol{x},\xi,t) \simeq f_0(\xi)(1 + f_1(\boldsymbol{x},\xi,t))\ , \tag{2.11}$$

$$\rho_w(\boldsymbol{x},t) \simeq \rho_{w0}(1 + \rho_{w1}(\boldsymbol{x},t))\ . \tag{2.12}$$

Moreover, making reference to [10], $\rho_{w1}(\boldsymbol{x},t)$ in (2.12) is given by:

$$\begin{aligned}\rho_{w1}(\boldsymbol{x},t) = &\sqrt{\pi}\tilde{w}_n(\boldsymbol{x},t) - \frac{2}{\pi}\int_{S^+} (\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y}))\ \times \\ &\times \frac{1}{r^4}\left(\int_0^\infty \rho w1\left(\boldsymbol{x}, t - \frac{r}{u}\right) u^3 \exp(-u^2)\, \mathrm{d}u\right) \mathrm{d}S\ + \\ &- \frac{4}{\pi}\int_{S^+} (\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y}))\ \times \\ &\times \frac{1}{r^5}\left(\int_0^\infty \boldsymbol{r}\cdot\tilde{\boldsymbol{w}}\left(\boldsymbol{x}, t - \frac{r}{u}\right) u^4 \exp(-u^2)\, \mathrm{d}u\right) \mathrm{d}S\ .\end{aligned} \tag{2.13}$$

In several applications MEMS response is proportional to the fundamental

mode $\boldsymbol{g}(\boldsymbol{x})$. As a consequence it is assumed that only a weak coupling between the structural response $q(t)$ and the fluid action exists, in the sense that fluid dynamics does not influence the shape of the displacement field $\boldsymbol{s}(\boldsymbol{x},t)$ or the velocity $\boldsymbol{w}(\boldsymbol{x},t)$, which are of the form:

$$\boldsymbol{s}(\boldsymbol{x},t)=\boldsymbol{g}(\boldsymbol{x})q(t),\quad \boldsymbol{w}(\boldsymbol{x},t)=\boldsymbol{g}(\boldsymbol{x})\dot{q}(t)\ . \tag{2.14}$$

Taking now the Fourier transform of eq.(2.13) and simplifying $\mathcal{F}(\dot{q})$ one has:

$$\begin{aligned} J(\boldsymbol{x},t)=&\sqrt{\pi}\tilde{\boldsymbol{g}}(\boldsymbol{x})\cdot\boldsymbol{n}(\boldsymbol{x})+\\ &-\frac{2}{\pi}\int_{S^+}J(\boldsymbol{y},t)(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y}))\frac{1}{r^4}T_3(i\tilde{\omega}r)\,\mathrm{d}S+\\ &+\frac{4}{\pi}\int_{S^+}(\mathbf{r}\cdot\tilde{\boldsymbol{g}}(\boldsymbol{y}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y}))\frac{1}{r^5}T_4(i\tilde{\omega}r)\,\mathrm{d}S\ , \end{aligned} \tag{2.15}$$

where quantities with tilde are normalized by a factor $\sqrt{2RT}$; $J(\boldsymbol{x})$ is the complex function such that the Fourier transform of ρ_{w1} is $\mathcal{F}(\rho_{w1})=J(\boldsymbol{x})\mathcal{F}(\dot{q})$ and

$$T_n(i\tilde{\omega}r)=\int_0^\infty u^n\exp\Big(-u^2-\frac{i\tilde{\omega}r}{u}\Big)\,\mathrm{d}u \tag{2.16}$$

are transcendental functions.

With a similar procedure an integral equation can be found for the pressure acting on every MEMS surface $\boldsymbol{p}(\boldsymbol{x})$. Thus by a post-processing of $J(\boldsymbol{x})$ one has:

$$\begin{aligned} \frac{\pi^{3/2}}{2\rho_0\mathcal{R}T}\boldsymbol{p}(\boldsymbol{x})=&\Big(1+\frac{1}{2}J(\boldsymbol{x})\Big)\frac{\pi^{3/2}}{2}\boldsymbol{n}+\pi\tilde{g}_n(\boldsymbol{x})\boldsymbol{n}(\boldsymbol{x})+\frac{\pi}{2}\tilde{\boldsymbol{g}}_t(\boldsymbol{x})+\\ &-\int_{S^+}\boldsymbol{r}J(\boldsymbol{y})(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y})\frac{1}{r^5}T_4(i\tilde{\omega}r)\,\mathrm{d}S+\\ &+2\int_{S^+}\boldsymbol{r}(\boldsymbol{r}\cdot\tilde{\boldsymbol{g}}(\boldsymbol{y}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{x}))(\boldsymbol{r}\cdot\boldsymbol{n}(\boldsymbol{y}))\frac{1}{r^6}T_5(i\tilde{\omega}r)\,\mathrm{d}S\ . \end{aligned} \tag{2.17}$$

The distribution $\boldsymbol{p}(\boldsymbol{x})$ is a very important parameter since it allows the calculation of the damping coefficient b and of the quality factor Q as a consequence. Indeed the equivalent damping term in the 1D reduced order model

(1.6) can be seen as:

$$b\dot{q} = \Big(\int_S \boldsymbol{g}(\boldsymbol{x}) \cdot \boldsymbol{p}(\boldsymbol{x}) \, \mathrm{d}S \Big) \dot{q} \ , \tag{2.18}$$

where the integral is extended over the whole surface S of the MEMS. The constant b can be conveniently expressed as:

$$b = \tilde{b}\rho_0 \sqrt{2\mathcal{R}T_0} = \tilde{b}p_0 \sqrt{\frac{2}{\mathcal{R}T_0}} \ , \tag{2.19}$$

where $\tilde{b}$ is a coefficient with the dimensions of a surface depending only on the problem geometry and the existing relation between density ρ_0 and pressure p_0 is $p_0 = \rho_0 \mathcal{R} T_0$.

Therefore by computing the integral equations (2.15) and (2.17), b can be calculated as (2.18) and thus Q is obtained straightforward (as explained in Sec.1.4). Details of the whole discussion presented so far can be found in [10] and [14].

Note that in equations (2.15) and (2.17) the trascendental functions $T_n(z)$ occur with different values of n. In order to evaluate them, one can proceed employing recursive formulas as explained in [15].

In the current implementation the asymptotic expansions are exploited:

$$T_n(z) \simeq \sqrt{\frac{\pi}{3}} 3^{-n/2} u^{n/2} \exp(-u) \sum_{k=0}^{\infty} \frac{a_k}{z^k} \ , \tag{2.20}$$

where

$$\begin{aligned}
u &= 3\left(\frac{z}{2}\right)^{2/3} \\
a_0 &= 1 \\
a_1 &= \frac{1}{12}(3n^2 + 3n - 1) \\
12(k+2)\ a_{k+2} &= -(12k^2 + 36k - 3n^2 - 3n + 25)\ a_{k+1} + \\
&\quad + \frac{1}{2}(n-2k)(2k+3-n)(2k+3+2n)\ a_k, \quad k \geq 0 \ .
\end{aligned}$$

First one starts generating T_1 according to the following relation:

$$2\ T_1(z) = \sum_{k=0}^{\infty} (a_k \ln z + b_k)\ z^k\ , \tag{2.21}$$

where

$$a_0 = a_1 = 0, \quad a_2 = -b_0, \quad b_0 = 1, \quad b_1 = -\sqrt{\pi}, \quad b_2 = \frac{3}{2}(1-\gamma),$$

$$a_k = \frac{-2a_{k-2}}{k(k-1)(k-2)}\ , \quad b_k = \frac{-2b_{k-2} - (3k^2 - 6k + 2)\ a_k}{k(k-1)(k-2)}\ .$$

Then T_0 and T_2 can be calculated knowing that

$$T_n^{'}(z) = -T_{n-1}$$

and

$$4\ T_2(0) = \sqrt{\pi}\ .$$

Finally all required T_n are generated by recursion according to

$$2\ T_n(z) = (n-1)T_{n-2} + z\ T_{n-3}, \quad \text{for } n \geq 3\ .$$

The calculation of asymptotic expansions can be avoided if the quasi-static condition holds, as explained in the following.

Quasi-static condition

It is worthwhile stressing that, at low frequencies, the transcendental functions $T_n(z)$ can be easily evaluated by means of the quasi-static condition.

Indeed, in many applications, MEMS are subject to sinusoidal electrostatic forces and the response of the MEMS can be computed through the Spring-Mass-Damper second order differential equation (1.6). The response itself consists of an oscillation with maximum amplitude A of the type $q(t) = A\sin(\omega t)$. The small perturbation condition $|\tilde{\boldsymbol{w}}| \ll 1$ can be expressed as $c_L \ll 1$, where coefficient c_L is given by $c_L = \tilde{\omega} A$. Moreover if L is the characteristic length of the flow (that is the maximum distance between two visible points), then

$c_Q = \tilde{\omega} L \ll 1$ represents the quasi-static condition.

This means that the time-of-flight of a molecule between two different surfaces is small with respect of the duration of one oscillation. If this condition is met the transcendental functions T_n can be evaluated in zero:

$$T_3(0) = \int_0^\infty u^3 \exp\Big(- u^2 \Big) \, \mathrm{d}u = \frac{1}{2} \ , \tag{2.22}$$

$$T_4(0) = \int_0^\infty u^4 \exp\Big(- u^2 \Big) \, \mathrm{d}u = \frac{3\sqrt{\pi}}{8} \ , \tag{2.23}$$

$$T_5(0) = \int_0^\infty u^5 \exp\Big(- u^2 \Big) \, \mathrm{d}u = 1 \ . \tag{2.24}$$

However, as already mentioned above, this assumption holds only at low resonance frequencies. Indeed, with increasing frequencies, the term $\tilde{\omega} L$ becomes comparable with the thermal velocity of molecules and the quasi-static condition is no longer satisfied. Thus, in order to have a model as general as possible, which holds also at high frequencies, in this work the implementation is carried out assuming small perturbation only, and so the T_n functions are evaluated through asymptotic expansions.

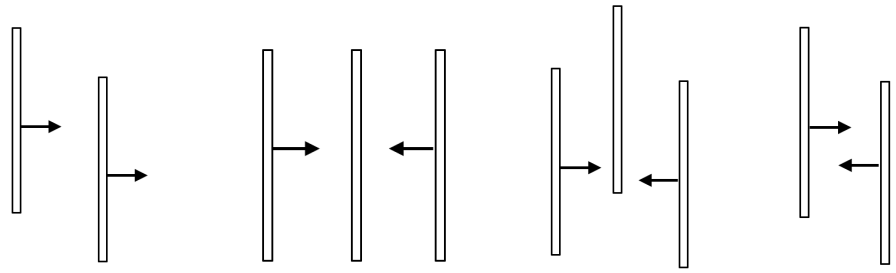
2.3 Numerical implementation

It can be noticed that equation (2.15) is very similar to the radiosity equation of Computer Graphics, which is a tool for the generation and manipulation of images on computer screens. One of the key elements is the presence of the visibility operator limiting the integration to the visible portion of surfaces. Basically, given a set of 3D objects and a viewing point, the objective is to determine which lines or surfaces of the objects are visible. Thus, inspired by this analogy, in [16] is implemented an efficient way to compute the domain of integration adapting some typical techniques of computer graphics, where the viewing points correspond to source objects.

Since due to technological constraints the vast majority of MEMS is composed by piecewise planar surfaces, the structure is initially schematized as a collection of non-overlapping large planar “parent” quadrangles. Each quadrangle is then meshed with “children” triangles using the software GMSH. In

the present implementation J is modelled as piecewise constant over each triangle. Equation (2.15) is then collocated at the centre of each triangle and the system of equations is solved by means of an iterative GMRS (Generalized Minimal Residual Method) solver. The final output of the simulation is the constant $\tilde{b}$ obtained through a post-processing of the force $\boldsymbol{p}(\boldsymbol{x})$; $\tilde{b}$ will be then exploited in order to compute the damping coefficent b as prescribed by the relation (2.19).

2.3.1 Testing the visibility condition



(a) Invisible. (b) Invisible. (c) Partially visible. (d) Totally visible.

Figure 2.2: Classification of surfaces according to their visibility.

In this section a description of the part of the code concerning the testing of the visibility of surfaces is provided. This part has to be fast and reliable in order to have a globally computationally efficient method.

Given a source point $\boldsymbol{x}$, the integral equations (2.15) and (2.17) must be limited to portion S^+ of the surface which is visible from $\boldsymbol{x}$.

Each quadrangle (and its triangles) is endowed with an outward normal vector $\boldsymbol{n}$ defining the positive sides of the surfaces. Visibility is tested only for the positive sides of quadrangles and eventually of triangles.

Two quadrangles can be *invisible* (Figg.2.2(a)-2.2(b)), *partially visible* (Fig.2.2(c)) or *totally visible* (Fig.2.2(d)). In the code a symmetric square matrix is built in order to keep a record of the mutual visibility of surfaces: entries are set to 0 if two quadrangles are invisible; set to 2 if they are totally visible and set to 1 if they are partially visible.

Since the number of quadrangles is much smaller than the number of triangles, establishing the visibility between quadrangles has a limited cost and visibility or invisibility directly transfers to the children. On the contrary if two parents are only partially visible, then visibility has to be tested for every child triangle with a much higher cost.

Thus the goal of this work has been that of improving the visibility part so as to reduce tests on triangles or, better, to completely avoid them.

The already implemented visibility part in the code (fully described in [17]) consists of 4 levels of tests: Level 1 and 2 detect the invisible surfaces; Level 3 looks for the totally visible surfaces; Level 4 performs visibility tests on triangles. In the following more detailed descriptions of these levels are provided.

Level 1: Hidden surface removal

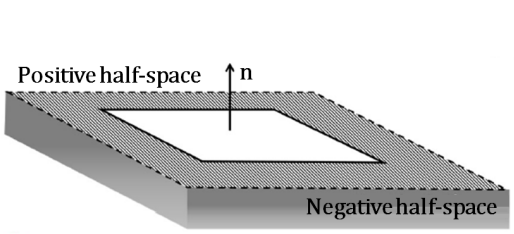
This first level rapidly detects quadrangles which are clearly invisible. Considering a given source quadrangle, its plane divides the space into two half-spaces according to its positive side, as shown in Fig.2.3(a).

If a target quadrangle has the four vertices in the negative half-space, then it is marked as invisible from the source and both entries of the visibility matrix are filled with zero.

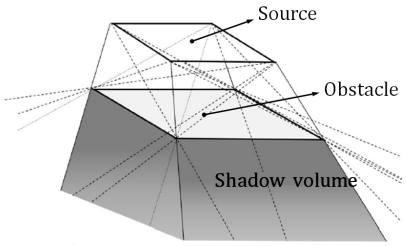
Level 2: Shadow volume

In this second level another test to identify invisible quadrangles is implemented. Those which have not been detected in the previous level are analyzed in this one using a shadow volume technique.

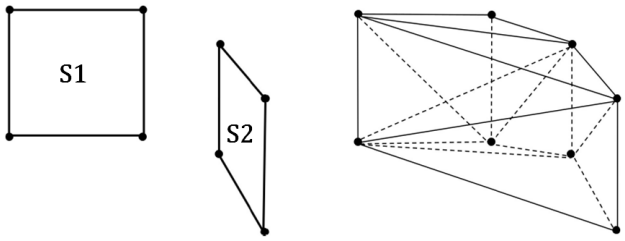
Considering a source quadrangle and a target quadrangle, obstacles (that is quadrangles between the source and the target) can hide the target from the source. In order to recognize obstacles, their shadow volume is built as follows: a ray is projected from each vertex of the source to each vertex of the obstacle. These projections will together form a set of four "pyramids", one for each source vertex, and their intersection is the total-shadow volume. See Fig.2.3(b). If a target has all the vertices inside the shadow volume, then it is



(a) Level 1: Hidden surface removal.



(b) Level 2: Shadow volume.



(c) Level 3: Delaunay triangulation.

Figure 2.3: Levels of the visibility test: Levels 1 and 2 detect invisible surfaces, while Level 3 looks for visible surfaces.

invisible to the source.

Level 3: Delaunay triangulation

In this third level a test to detect if two quadrangles are totally visible is implemented. Thus those quadrangles which have not been identified as invisible in the first two phases are addressed in this level.

An algorithm based on 3D Delaunay triangulation (or tetrahedralization) is implemented. Triangulation of a given set P of n points in 3-dimensional space is a decomposition into tetrahedra of the convex hull of P such that:

1. vertices of tetrahedra belong to P;

2. intersection of two tetrahedra is either a vertex or an edge or a face.

Given a source quadrangle S_1 and a target quadrangle S_2, the set P is populated with their 8 vertices and the Delaunay triangulation is built, as depicted in Fig.2.3(c). If none of the other quadrangles intersect the convex hull, then

the source and the target are fully visible and their entries in the visibility matrix are set to 2.

Level 4: Triangles visibility

All the quadrangles not identified are considered partially visible (thus their entries are set to 1) and the visibility is consequently tested on their triangles by means of a point-to-point test visibility. Two triangles are said to be *visible* if the segment connecting their centers of mass does not encounter obstacles. As already mentioned, since the number of triangles is high, visibility test on triangles is time consuming, thus computationally inefficient.

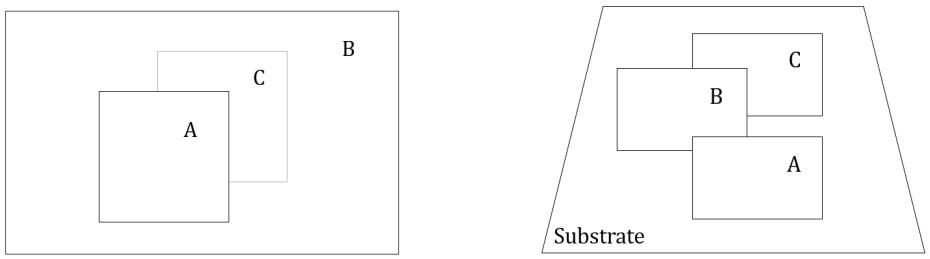
The goal of the thesis is to speed up the total visibility computation of surfaces. Thus in the next chapter the improvements introduced in the code are presented.

2.4 Bugs in the previous implementation

Besides the low speed which characterizes the test in Level 4, it is worthwhile to briefly explain some bugs found in the original implementation of the code which were identified and fixed during the thesis.

2.4.1 A false-positive in the identification of visible quadrangles

The first discovered problem concerns an imperfect implementation of the test in Level 3. In the original implementation, after having built Delaunay's convex volume between a target and a source surface, it is only verified that triangular boundary faces of the volume do not intersect other quadrangles. This control has been conceived thinking about wide surfaces which evidently are obstacles between the source and the target, but whose vertices and sides do not intersect the volume, as depicted in Fig.2.4(a) where A and C are the source and the target quadrangles and B is the wide obstacle between them. If this control had not been implemented, source and target quadrangles would



(a) (b)

Figure 2.4: (a): a wide surface B completely covers surfaces A and C. (b) Example of configuration where surfaces A and C are wrongly detected as "totally visible" in Levels 3 though B partially covers them.

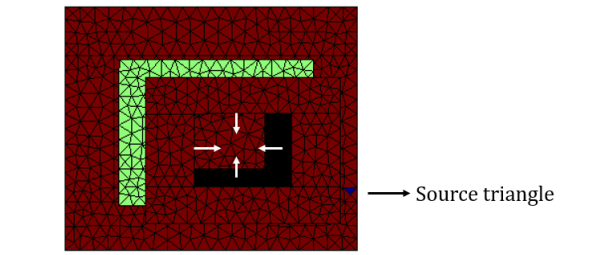
be marked as "totally visible" while they are not.

However such implementation is not complete since it does not take into account situations where boundary sides of other quadrangles intersect the Delaunay's volume, that is cases where an obstacle (B) partially covers two quadrangles (A and C), see Fig.2.4(b). The lack of the implementation of this second control leads to false-positive visible quadrangles.

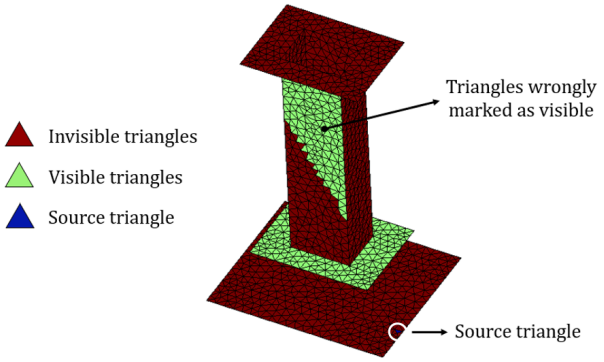
2.4.2 Change of status for triangles

Contrary to quadrangles case, in the original implementation of the code any symmetric matrix has been built in order to keep a record on triangles visibility. Because of the lack of such data structure, if a target triangle T_i results to be invisible to a source triangle T_j, then they are both marked as "invisible" one to the other. However when the target triangle T_i becomes the source one, its visibility against all the other triangles is tested, including the ex-source triangle T_j which now is a target triangle. Therefore with a second control on the same two triangles with inverted roles, their status may change from "invisible" to "totally visible".

In Fig.2.5(b) is reported an example of what highlighted so far: the green triangles at the top of the hole were classified as "invisible" to triangles of the substrate (at the bottom of the hole) and thus coloured in red; but if a triangle of the bottom becomes a source triangle (coloured in blue in the picture) a



(a)



(b)

Figure 2.5: Example of geometry where some triangles wrongly changed their status to "visible", when a triangle goes from "source triangle" to "tested triangle" and viceversa.

wrong computation of the scalar product between their normal vectors leads to a wrong result: triangles at the top of the hole, already classified, are detected as completely visible by those on the substrate and their attribute is changed from "invisible" to "visible" (thus re-coloured in green).

2.5 Failure in identifying some invisible quadrangles

Besides the two bugs just introduced, the old implementation suffers from a big imprecision: the non-identification of clearly invisible surfaces with the two tests in Level 1 and Level 2. As an example, making reference to Fig.2.6, quadrangle D is completely hidden by B and C to the sight of A. However D

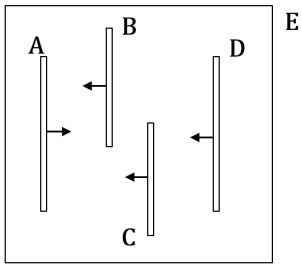


Figure 2.6: Top view of an example of geometry where an invisible surface (D) is not recognized by tests in Levels 1 and 2.

has all of its four vertices in the positive half-space of source surface A, thus it passes test in Level 1 (Hidden Surface Removal). Moreover, in Level 2, the total shadow volume is built considering one obstacle surface at a time. By doing so it is evident that, once again, surface D does not have all of its four vertices either in the shadow volume generated by B or in that generated by C. Thus it passes also Level 2 without being recognized as invisible surface.

Though tests in the first two levels are correctly implemented, there are situations, as the one just highlighted, where quadrangles are not recognized as invisible by the code. This is not properly an error, since not-identified surfaces will be marked with 1 (partially/potentially visible) in the visibility matrix and processed at triangles level and marked as "invisible" later. However test on triangles is time consuming and affects code performance, as already mentioned, and therefore should be avoided. Thus if a target surface is clearly not partially visible, then it would be better if it is detected as "invisible" with the first shot.

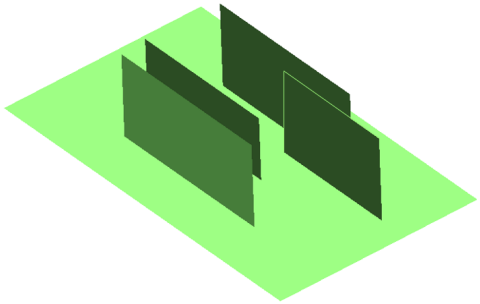
2.6 Structures geometry

As mentioned in Sec.2.3, the vast majority of MEMS is composed by piecewise planar surfaces. Thus the elementary structural element is a quadrangular surface. The combination of quadrangles results in more complicated geometries: comb fingers, parallel plates and holed mass. Code performance are tested over each of these configurations: for each of them an input file on GMSH is created.

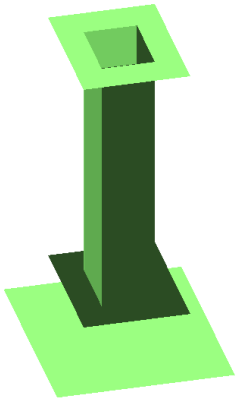
In Fig.2.7 the just introduced geometries are shown (from the simplest to the most complicated):

- parallel surfaces with a substrate: a very simple configuration used to verify if the code works fine in trivial situations (Fig.2.7(a));
- hole: since studied devices in this work are constituted by a tilting holed mass, a schematization of a single hole is needed (Fig.2.7(b));
- comb fingers (Fig.2.7(c));
- parallel plates (Fig.2.7(d));
- three holes in a row: each hole of the tilting mass is said to be a Level 1 hole, a Level 2 hole or a Level 3 hole according to its distance from the boundary of the mass. Thus, in order to take into account boundary effects, distinguishing between holes is useful. In Fig.2.7(e) the structure of three different levels holes is schematized.

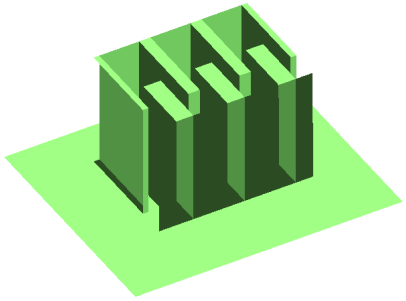
Solutions implemented to fix the bugs introduced so far are presented in the following chapter. Besides performance of the new implementation are provided as well.



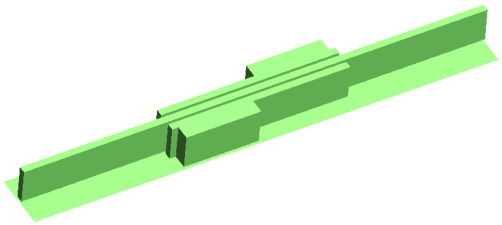
(a) Parallel surfaces on a substrate.



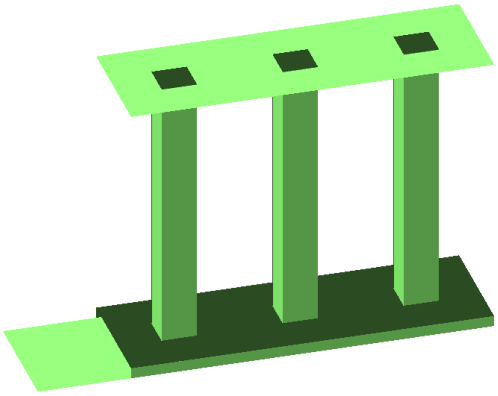
(b) Single hole of a holed mass.



(c) Comb fingers.



(d) Parallel plates.



(e) Three holes in a row.

Figure 2.7: Structures geometry used as inputs of the code.

Chapter 3

Numerical model: implemented improvements

As already mentioned in the previous chapter, the goal of this work is to develop an efficient model to estimate the damping coefficient b, in order to predict devices performance.

While trying to improve the existing code, some bugs, introduced in Sec.2.4, have been found. Thus at first the focus has been on fixing these bugs.

3.1 Improvements on the code: fixing of the bugs

3.1.1 False-positives in the identification of visible quadrangles

Let us start dealing with the first bug: false-positives in visible surfaces detection. In the original implementation, after the built of Delaunay's triangulation (Level 3) between two quadrangles, the sides of the boundary faces of the volume are given as inputs to the *vol_boundary* routine. This routine is exploited to control if some of these faces intersect other quadrangular surfaces, but such an implementation does not take into account situations where other quadrangles partially cover the source and the target ones, as explained in Sec.2.4.1.

To avoid this problem, the new implementation considers all the vertices used to build Delaunay's volume and, for each pair of vertices, it traces a segment; then, by means of the routine *intrett*, it is verified whether this segment intersects another quadrangular surface or not. If so, controls between the current source and target surfaces interrupt and the two quadrangles are marked as "partially visible" by filling their entries in the surfaces visibility matrix with 1. If controls have a negative outcome, then the two surfaces are considered "totally visible" and their entries are set to 2.
Note that the *vol_boundary* and the *intrett* routines belong to a library, developed by Joe in [18], which is available online at [19].

3.1.2 Change of status for triangles

In order to fix the second bug, presented in Sec.2.4.2, it is worthwhile noticing that the change of status in the mutual visibility of two triangles is caused by a wrong calculation of the scalar product between their normal unit vectors. Thus they can be wrongly not recognized as invisible (see Fig.2.5(b)). To avoid this problem a different test, similar to the one in Level 1 for surfaces, has been implemented: Hidden Elements Removal. The new routine *hiddenelems* takes as input two triangles, a source triangle T_i and a target one T_j; it computes the plane where the source lays and, according to the direction of its normal vector, considers if the target T_j has all of its three vertices in the negative semi-half space of the source or not. If so, both the target and the source are marked as "invisible". An excerpt of this routine is shown in Fig.3.1. A loop is performed on the vertices of the test triangle. Then for each vertex its coordinates are stored in the record `point` and the parameter `checkp` is the scalar product with the normal vector `n` of the plane, where the source triangle lays. Then `checkp` is checked: if it is negative or at least null, the target triangle stands behind the source one or at least in its same plane. Thus, if all the three vertices provide the same result for the variable `checkp`, then the boolean `inv` is set to 1 and the two triangles are effectively marked as invisible.

Besides the implementation of this new routine, to prevent that the same pair of triangles is tested twice with inverted roles, a data structure similar

```
inv = 0

n = list_subsurface(subsup1)%normal
d = list_subsurface(subsup1)%d

count = 0   ! counts how many points are behind the element i1
count1 = 0  ! counts how many points are in the plane of the element i1

do i3 = 1,3

    node = list_elements(el2)%nodes(i3)
    point = list_nodes(node)%coordinates(:)

    checkp = n(1)*point(1) + n(2)*point(2) + n(3)*point(3) + d

    if(checkp<-1.d-5) then   !<0
        count = count + 1
    elseif(checkp<1.d-5) then   !=0
        count1 = count1 + 1
    endif

enddo

if(count+count1==3) then
    ! element i2 results to be invisible to element i1
    inv = 1
endif
```

Figure 3.1: Excerpt of the *hiddenelems* routine, where the main parameter, which discriminates if a test triangle is behind or on the same plane of the source triangle, is highlighted.

to the one for quadrangles has been introduced: a symmetric square matrix which keeps a record of the mutual visibility between triangles. Indeed, thanks to the symmetry of the matrix, once visibility of triangles T_i and T_j has been verified, both entries m_{ij} and m_{ji} of the visibility matrix are filled with 0 (invisible triangles) or 1 (partially visible triangles).

This solution performs a first examination on triangles which helps in reducing the number of elements that must be processed in the point-to-point test of Level 4 (see Sec.2.3.1).

3.2 Improvements on the code: overcoming of the imprecisions

As already declared, the previous version of the code fails in the identification of clearly invisible quadrangles, which leads to the processing of those missed invisible surfaces at triangles level. As already stressed more than once, controls on triangles are highly time consuming and must be avoided or at least reduced. Although this failure is rather an imprecision than an error, since it does not provide a wrong result in the classification of quadrangles, it has to be fixed as well as the bugs analyzed and solved above.

Therefore in this section the proposed solution, which mainly contributes in improving the performance of the original code, is described.

3.2.1 Failure in identifying some invisible quadrangles

The idea which solves this imprecision and, at the same time, reduces the number of triangles that must be processed with the point-to-point test in Level 4, is that of partitioning the starting surfaces in four sub-surfaces. In this way, all the quadrangles which have not been identified as invisible or totally visible at the end of the test in Level 3, are split in four in the new routine *freeB_splitting_sup* (better described in the following). Then tests in Levels 1, 2 and 3 are performed again on each "children" quadrangle.

By splitting surfaces, smaller portions of them are processed in the visibility tests. For an unidentified surface (that is for a surface which is not classified as "invisible" or "totally visible" and therefore is marked as "partially visible") this process makes easier the detection of which portion of it is totally visible and which is instead invisible from a given point of view.

Note that visibility of surfaces is inherited by their triangles, therefore classifying as much quadrangles as possible helps in reducing the number of triangles to be classified later.

Moreover the procedure of splitting the surfaces and processing the new quadrangles does not require much time, thus it is even more preferable to the point-to-point test on triangles performed in Level 4.

A brief description of the working of this routine is now provided.

A close up of the splitting routine

In the new implemented routine, as already mentioned above, quadrangular surfaces are partitioned in four.

At first, each entries of the visibility matrix for surfaces is analyzed by rows: at the first occurrence of a 1, surface related to that row is partitioned by the routine and the scan of the matrix goes on analyzing the following row. Indeed, as already explained in the previous chapter, since the matrix is a square one, founding at least a 1 in a row means that the surface under consideration is partially visible to another surface and viceversa. Thus it needs to be split.

Once a surface-to-be-split is detected, it is passed as input to a subroutine named *split* which, starting from the properties of the surface, computes all the parameters useful when performing the visibility tests on its "children" subsurfaces (such as vertices, barycentres, normal unit vectors, etc.).

After the surface is partitioned into its four subsurfaces, each of its elements must be assigned to one of the "children" quadrangles and this is accomplished by means of the subroutine *els_partition*.

The exploited logic to decide how to assign triangles to the subsurfaces, considers the barycentre of the triangle under analysis and all the barycentres of the four subsurfaces. Then for each couple barycentre (of the triangle) - barycentre (of a subsurface) it computes the distance in between. The couple with the lowest distance is selected and the element is assigned to the subsurface which the selected barycentre belongs.

An excerpt of the code of the splitting routine is depicted in Fig.3.2: where two "do" loops are performed in order to scan the visibility matrix `sup_visible` by rows. When an entry 1 is found, the code enters the "if" condition and the surface is split in four. At the end of the two loops, the already introduced *els_partition* routine is called.

As mentioned above (and confirmed in the next section), the procedure of splitting surfaces does not require much time, therefore it has been thought also to iterate the splitting until the area of a subsurface would have been

```
do i1=1,number_surface

! store for each surface its barycentre
! split it in 4 by exploiting coordinates of its barycentre
! repeat the algorithm

    do i2=1,number_surface
        if(sup_visible(i1,i2)==1)then

            ! surface i1 is partitioned
            ! since sup_visible is a square matrix, surface i2 is split as well subsequently
            do nsubsup=1,4
                count = count+1
                call split(i1,count,nsubsup)
                if( count > number_subsurface) then
                    write (*,*) 'error: in splitting_sup, id subsurface > num_subsurface'
                    exit
                endif
            enddo
            exit
        endif
    enddo

enddo

! assign each element of the master surface to each of its subsurfaces
call els_partition
```

Figure 3.2: Excerpt of the code of the splitting routine: the calls to the subroutines *split* and *els_partition* are highlighted in yellow.

comparable to that of the triangles of the mesh. However such idea is not reasonable in terms of occupied memory: indeed consider, for example, the simplest structure geometry constituted by four parallel surfaces, depicted in Fig.2.7(a). By splitting each of them into four sub-surfaces one obtains 16 new quadrangles. Assuming that, after the visibility tests, half of the sub-surfaces remains of unknown visibility, then, at the second iteration of the splitting procedure, there would be 8 sub-surfaces to be split in 4, with an amount of 32 total new sub-surfaces. Since the data structure which keeps track of the mutual visibility of the surfaces is a square matrix, one would have a 32×32 matrix, which results to be unaffordable.

Nonetheless, with only one subdivision of the surfaces, the obtained results are satisfactory in terms of saved computational time, as shown in the following.

3.3 Performance of the new implementation

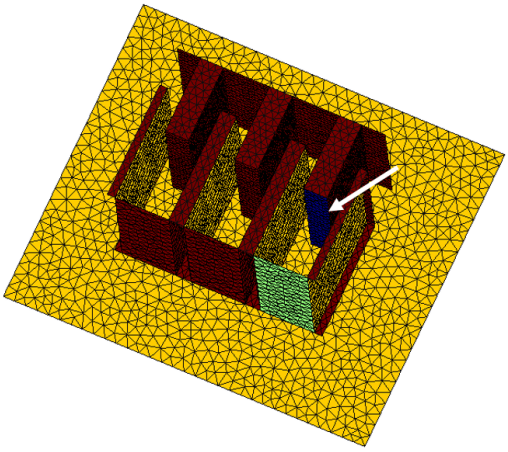
To better clarify, the whole new implemented procedure can be summarized as follows:

1. tests in Levels 1 (Hidden Surface Removal), 2 (Total Shadow Volume) and 3 (Delaunay's triangulation) are performed on quadrangles;
2. all the unidentified surfaces, that is the partially visible quadrangles characterized by entries 1 in the visibility matrix, are partitioned by means of the routine *freeB_splitting_sup*;
3. their "children" sub-quadrangles are processed again with tests in Levels 1, 2 and 3;
4. once sub-surfaces have been analyzed, triangles of those portions of quadrangles still unidentified are addressed in the Hidden Elements Removal test;
5. finally, the point-to-point test in Level 4, described in Sec.2.3.1, is performed on the remaining triangles still unclassified.

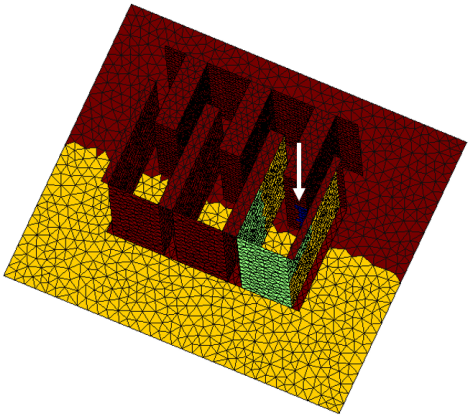
3.3.1 Visibility performance

In the figure above, evidence of the reduction of the number of triangles to be processed in the point-to-point test with the new implemented routines is shown. Invisible surfaces are coloured in red, totally visible surfaces are coloured in green, partially visible surfaces are coloured in yellow, while the source surface (the point of view) is the one in blue.

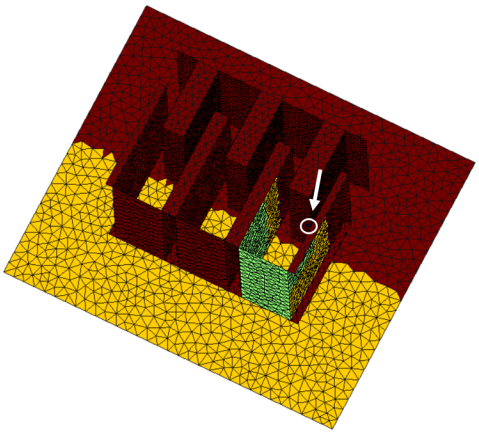
Consider the comb fingers geometry as an example. After the three tests on surfaces, some quadrangles constituting the walls of the comb and the entire substrate remain unclassified, thus coloured in yellow. In the original code, triangles belonging to yellow surfaces are directly addressed in the point-to-point test, but, as one can see in Fig.3.3(a), without any further analysis on them, the total amount of yellow triangles is large.



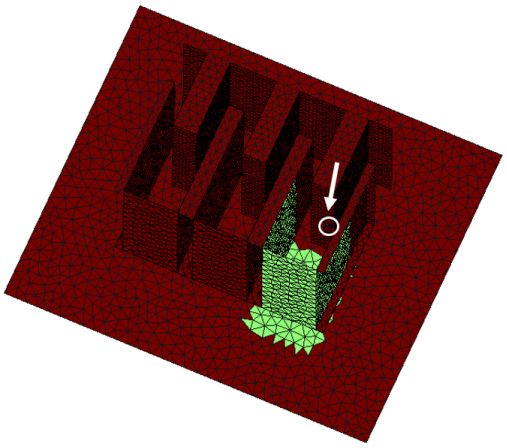
(a) Output of the original implementation after tests on surfaces (Levels 1, 2 and 3).



(b) Output after tests on partitioned quadrangles (*freeB_splitting_sup* + Levels 1, 2 and 3).



(c) Output after Hidden Triangles Removal (*hiddenelems*).



(d) Final output after point-to-point test (Level 4).

Figure 3.3: Steps of the surfaces visibility. For each step, the source surface or the source triangle is highlighted in white.

With the implementation of the splitting routine, half of the substrate and some walls of the comb are detected as "invisible" and are coloured in red; while the remaining walls of the comb have a totally visible portion which is coloured in green. It is clearly evident that, in this case, the number of yellow (unclassified) triangles is lower than in the previous step. See Fig.3.3(b).

After having subdivided the surfaces and having processed the resulting sub-surfaces, the Hidden Elements Removal test is performed and its output is shown in Fig.3.3(c). Some triangles on each of the two remaining yellow portion of comb walls are now detected as "invisible" and coloured in red.

The remaining yellow triangles are then processed in the point-to-point test in order to have them, once for all, classified as "totally visible" or "invisible". The final output is shown in Fig.3.3(d).

It is worthwhile stressing again that the old code directly proceeds with the point-to-point test on triangles after the visibility tests on surfaces. Thus from the situation depicted in Fig.3.3(a) it directly goes to the situation illustrated in Fig.3.3(d), without any intermediate step. This is inconvenient since it affects time performance. The new code, instead, passes through those intermediate steps (Figg.3.3(b)-3.3(c)) saving computational time. Evidence of the time performance mentioned so far are provided in the following paragraph.

Finally, note that the outcomes of the visibility tests of the improved implementation are more accurate and always correct than those of the old code, since all the bugs have been fixed and improvements have been made.

3.3.2 Time performance

In this paragraph comparisons between the time performance of the two implementations are provided.

Prior to proceed with the evaluation of the overall computational time spent by the two codes, the timing on each block of the two codes is performed, so as to confirm that the routine performing tests on elements is the one which takes longer.

Times of each block of the old code are stored in Tab.3.1, which has on the rows the geometries presented in Sec.2.6, from the simplest to the most

complicated, and on the columns the considered blocks. Thus on the first column is reported the overall time taken by the three routines of the first three levels, which perform visibility tests on surfaces; while on the second column is reported the time spent by the only routine of Level 4, which performs the test on elements.

Table 3.1: Computational time taken by each block of the old code, for each of the five considered geometries.

	Visibility on surfaces (Levels 1, 2, 3)	**Visibility on elements** (Level 4)
Parallel Surfaces	$< 1\ ms$	$0.078\ s$
Hole	$0.016\ s$	$0.875\ s$
Comb Fingers	$0.047\ s$	$19.391\ s$
Parallel Plates	$0.078\ s$	$36.062\ s$
3 Holes	$0.031\ s$	$28.656\ s$

This preliminary test confirms what stated above: the test in Level 4 on triangles is the one demanding the largest computational time.

The same measurements are also performed on the new code and the results are reported in Tab.3.2.

Comparing now values in the two tables, one can notice that timing of tests on quadrangles (first column in both tables) provides the same results for both the implementations, as expected. Besides the timing of tests on triangles is higher in the old implementation, than in the new one (last column in both tables). Indeed intermediate steps (second and third columns in Tab.3.2) implemented in the new model help in saving computational time.

Moreover it is worthwhile observing that the *freeB_splitting_sup* routine is truly fast, therefore the splitting procedure has revealed to be a highly convenient introduction in the code. At last one can notice also that timing of the visibility test on elements, in the case of the simplest geometry (Parallel

Table 3.2: Comutational time taken by each block of the new code, for each of the five considered geometry.

	Visibility on surfaces (Levels 1, 2, 3)	**Splitting of surfaces**	**Visibility on subsurfaces** (Levels 1, 2, 3)	**Visibility on elements** (H.E.R + Level 4)
Parallel Surfaces	$< 1\ ms$	$< 1\ ms$	$< 1\ ms$	$0.078\ s$
Hole	$0.016\ s$	$< 1\ ms$	$0.031\ s$	$0.594\ s$
Comb Fingers	$0.047\ s$	$< 1\ ms$	$0.281\ s$	$10.9\ s$
Parallel Plates	$0.078\ s$	$< 1\ ms$	$0.828\ s$	$17.5\ s$
3 Holes	$0.031\ s$	$< 1\ ms$	$0.281\ s$	$25\ s$

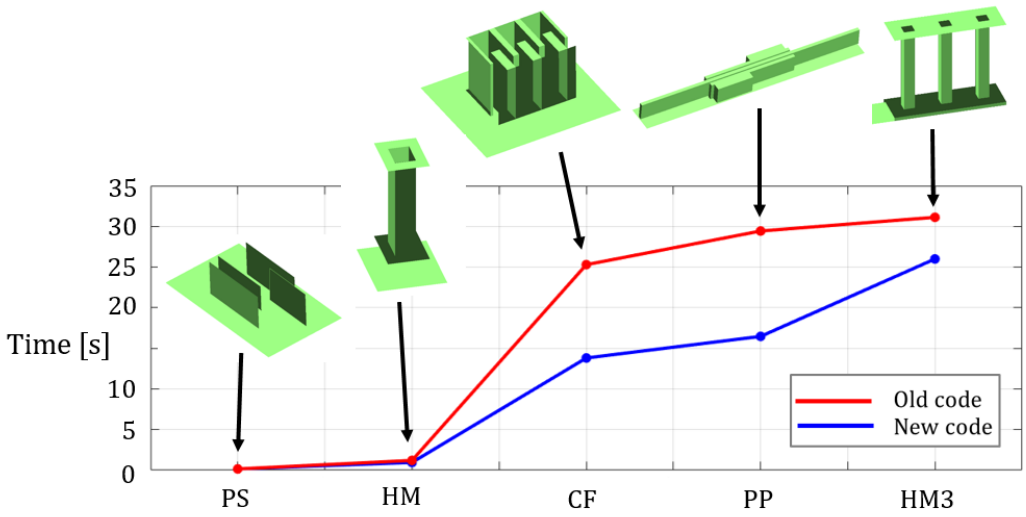
Surfaces, corresponding to first row in each tables), provides the same identical result. This is not surprising since the concerned geometry represents a trivial case.

General comparisons on the overall performance of the two code are then performed and in the following one can find the results.

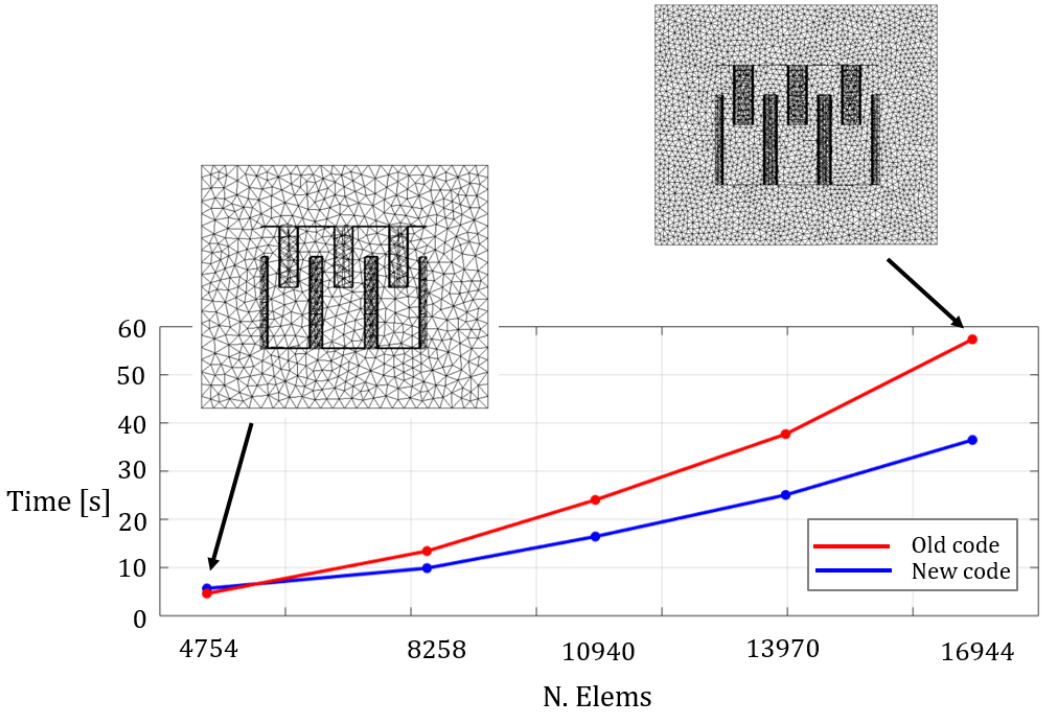
First the total amount of computational time spent by the two implementations, with all the structures geometry (introduced in Sec.2.6), is compared. Making reference to Fig.3.4(a), time is reported on the $y-$axis, while the type of tested geometry is reported on the $x-$axis. The red line represents the time spent by the old implementation, while the blue line shows the computational time required by the new implementation.

As one can see, for simple geometries (parallel surfaces and single hole) the two implementations take approximately the same time. For more complex geometries, the required computational time of the new implementation is about half of the time required by the old implementation.

A second comparison is provided considering mesh with varying number of elements. As an example the comb finger geometry is taken into account, since it is neither too simple nor too complex. Then the number of triangles which constitutes the mesh has been changed from approximately 5000 (coarse mesh) to 17000 (refined mesh) elements. In Fig.3.4(b) one can notice that, as expected, with the number of elements in the mesh growing larger ($x-$axis),



(a) Comparison between simulation times of the old (red) and the new (blue) code with increasing geometry complexity.



(b) Comparison between simulation times with increasing number of elements in the mesh.

Figure 3.4: Comparisons between time performance of the old code with bugs (red) and the new and improved one (blue).

the required computational time to compute the visibility of the geometry increases ($y-$axis). However, the new proposed implementation always takes less time than the old implementation. Especially with more refined mesh, the gap between the computational time of the two implementations becomes relevant.

In conclusion, the proposed modified code results to have better time performance both with increasing complexity in the geometry, but fixed number of elements in the mesh, and with fixed type of geometry, but varying number of triangles in the mesh.

All the tests on time performance have been run on a computer with an Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz processor, 8 GB RAM and Windows 10 as operating system.

Chapter 4

Experimental validation

In this chapter are presented the experimental results which validate the new model for the prediction of the damping coefficient b for MEMS operating in the fluid damping regime in near-vacuum and in a range of frequencies going from 10 kHz to higher frequencies, up to 100 kHz. Such frequency range is expected to be of interest for next-generation inertial sensors. Besides, with increasing frequency, the "quasi-static" approximation (presented in Sec.2.2) gradually becomes invalid, hence the need for new modelling tools. The proposed method has been validated through eight different structures with varying resonance frequency and air gap.

4.1 Test structures

The test structures, shown in Fig.4.1, are formed by a holed proof mass with area equal to $(270)^2 \mu m^2$, anchored through two torsional springs attached to the lower side, whose out-of-plane thickness is 30 μm.

This basic structure is replicated eight times, according to the four different target frequencies (10 kHz, 20 kHz, 50 kHz and 100 kHz) and the two vertical gaps (1.2 μm and 1.8 μm).

The difference in the resonance frequency is ensured by the different geometry of the torsional springs. As one can notice in the figure, from top to bottom, springs gradually becomes shorter and thicker. Thus devices whose

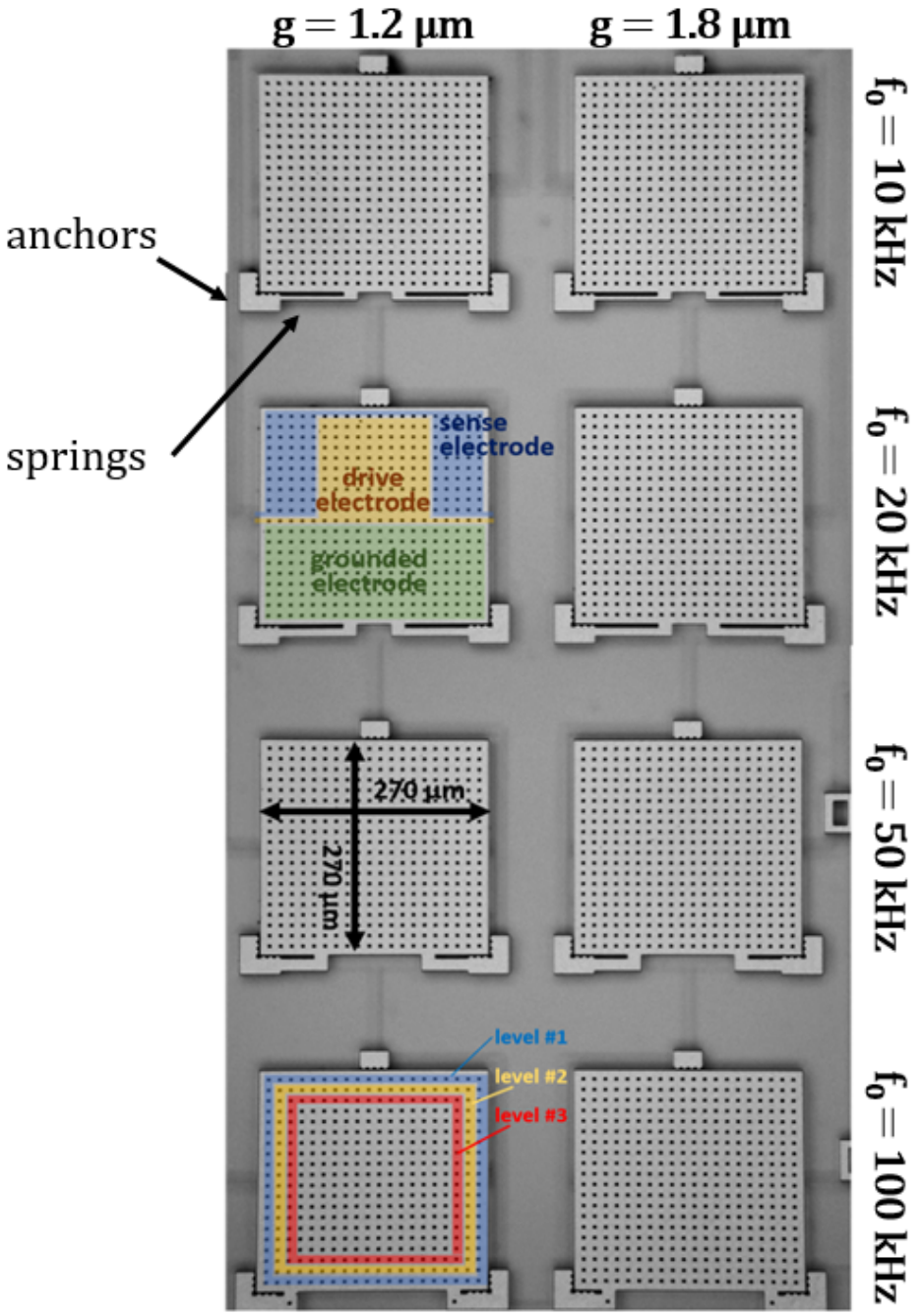


Figure 4.1: Optical top view of the 30-μm-thick test structures fabricated to validate the proposed damping modelling. Rows represent varying frequency, columns represent different vertical gaps.

resonance frequency is equal to 10 kHz are characterized by the thinnest and longest springs, while devices vibrating at 100 kHz have the thickest and shortest springs.

The eight structures are encapsulated in a package, into which the pressure is nominally 0.7 mbar. One can actuate the structures and acquire the response signal by means of access pads on the package.

Each structure can be electrostatically actuated on its first torsional resonant mode, and capacitively sensed through electrodes designed underneath the proof mass.

Prior to start measurements, it has been necessary the calculation of the pull-in potential of each structure. That is, the maximum allowed potential which can be used to actuate the proof mass, without incurring in a mechanical instability that can damage the device. Indeed, if the provided voltage is too high, the proof mass will displace too much from its rest position colliding with the package walls. The pull-in voltage is given by the relation:

$$V_{\text{pull-in}} = \sqrt{\frac{kg^3}{2\varepsilon_0 A_{\text{eq}} d^2}} \ , \tag{4.1}$$

where:

- $k = \omega_0^2 I$ is the springs stiffness, where $\omega_0 = 2\pi f_0$ and I is the momentum of inertia;
- g is the vertical gap $[m]$;
- $\varepsilon_0 = 8.854 \cdot 10^{-12}$ is the vacuum permittivity $[F \cdot m^{-1}]$;
- d is the distance between the center of mass and the rotation axis of the structure $[m]$;
- $A_{\text{eq}} = A_{\text{tot}} - 400 \cdot A_{\text{hole}}$ is the equivalent area of the holed mass $[m^2]$, where factor 400 represents the total number of holes on each proof mass.

In Tab.4.1 the calculated pull-in voltages are indicated.

Table 4.1: Pull-in voltages of each type of structure.

	$\mathbf{V}_{\mathrm{pull-in}}$ [V]	
Frequency [kHz]	**gap = 1.2 μm**	**gap = 1.8 μm**
10	7	13
20	14	26
50	35	64
100	70	129

4.2 Numerical results on damping estimation

As already highlighted in Sec.2.2, the existing "quasi-static" formulation, implemented in [20]-[21], is based on the assumption that the time-of-flight of a molecule between two different surfaces of the structures is small with respect to the duration of one oscillation. If this condition is met and if other sources of dissipation can be neglected, the quality factor Q is expected to have a linear evolution as a function of the working frequency.

However, as the frequency f increases, its product $f \cdot L$ with a typical dimension of the flow becomes comparable to the thermal velocity of molecules: $(2RT_0)^{1/2}$. Therefore it is adopted the formulation put forward in [10] and explained in Sec.2.1. The new proposed model can be considered exact and fully 3D and it makes the only assumption of small perturbation (see Sec.2.2).

For modelling purposes and numerical damping estimation, the MEMS structure is split into elementary units, already presented in Sec.2.6. In this case, the elementary units for the holed mass are formed by a vertical hole an its boundary pitch, with the electrode underneath.

Note that holes are classified into levels according to their distance from the proof mass. As explained in [21], cells are organized in levels in order to account for edge effects. According to this classification, level 1 cells are those closest to the boundary, while cells with levels larger than 3 are considered to be at regime and treated imposing perfectly periodic boundary conditions. In Fig.4.1 different hole levels are highlighted on a structure: holes in level 1 are coloured in blue, holes in level 2 are yellow and holes in level 3 are red.

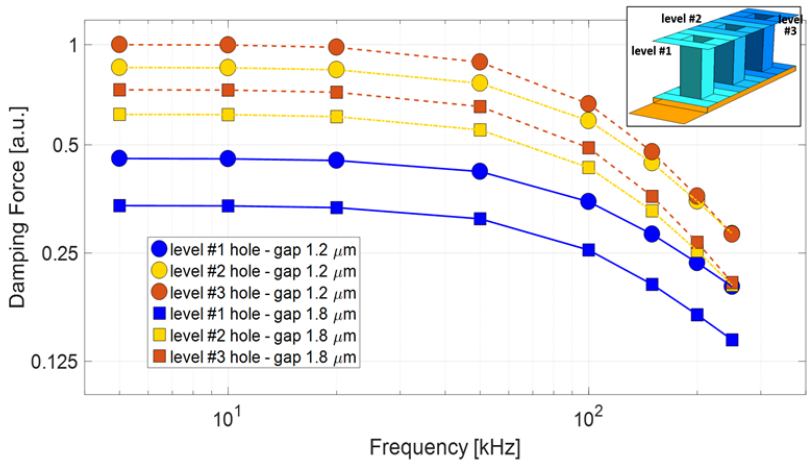


Figure 4.2: Predicted damping force on elementary holed blocks as a function of frequency and gap.

In Fig.4.2 are reported simulation results of the damping force acting on each hole, for different hole levels.

The numerical predictions of damping are reported for both the gap values (1.2 μm represented by circles, 1.8 μm represented by squares). It is clearly shown that damping force depends on the frequency, especially from 50 kHz frequencies and higher: damping b results to decrease more than linearly with increasing frequency, therefore, since b and Q are inversely proportional (see Sec.1.4), it is expected that the quality factor Q increases more than linearly with increasing frequency.

For the sake of completeness, other possible sources of dissipation have been considered for high frequencies. Anchor losses have been simulated as suggested in [22]; according to obtained estimations, they can be safely neglected since $Q_{\text{anc}} > 10^5$ for all the considered configurations. Thermoelastic analysis have also been performed, with the following material parameters:

- coefficient of linear thermal expansion $\alpha = 2.6 \cdot 10^{-6}\ K^{-1}$;
- thermal conductivity $\lambda = 148\ W/(m \cdot K)$;
- heat capacitance $C_p = 700\ J/(kg \cdot K)$.

The thermoelastic quality factor Q_{ted} is mainly associated to bending in the anchors and decreases with increasing frequency (stiffer springs). At 100 kHz, one gets a $Q_{\text{ted}} = 46000$ which can be thus neglected as well. Indeed, as already explained in Chapter 1, the overall quality factor can be calculated as:

$$Q = \frac{1}{1/Q_{\text{fluid}} + 1/Q_{\text{ted}} + 1/Q_{\text{anc}}} \ , \tag{4.2}$$

therefore, since Q_{ted} and Q_{anc} have high values, relation 4.2 can be approximated as

$$Q \approx Q_{\text{fluid}} \ . \tag{4.3}$$

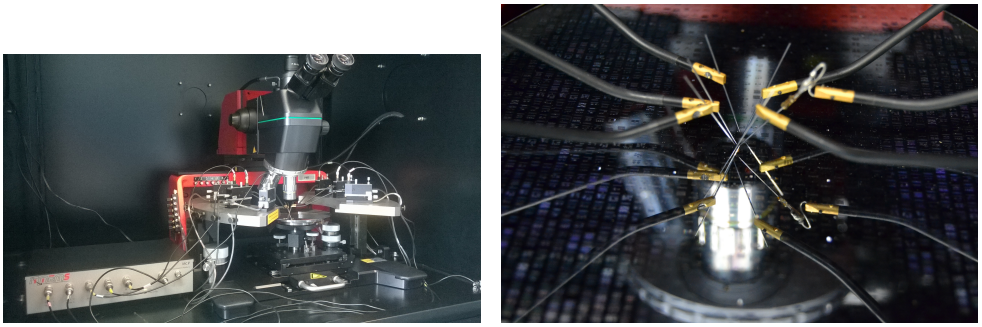
Thus the main contribution to Q is given by the fluid quality factor.

4.3 Description of the instrumentation

The experimental setup is based on a probe station, a MEMS characterization platform (MCP-G by *ITmems*) and a waveform generator (33250a by *Agilent*).

Tests on MEMS devices have been carried out, at first, on the probe station in the clean room of the Laboratory of MEMS and microsensors, Politecnico di Milano. It consists mainly in microscopic needles (shown in Fig.4.3(b)) which easily contacts the access pads of a package in order to actuate the MEMS in it and acquire their signals. Actuation and acquisition are made with the help of the MCP. For this purpose, the probe station is attached to it by means of cables (as depicted in Fig.4.3(a), where the MCP is on the left and the probe station is on the right).

With the help of the MCP, one can study the dynamics of MEMS by tracking the capacitance variations of their moving parts (rotors) with respect to those at rest (stators). MEMS capacitances are measured through a sinusoidal probe-signal of frequency much higher than the resonance frequency of the device under test. By doing so, the motion of the MEMS is not perturbed. Then the unknown capacitance is determined by measuring the current $i_C(t)$



(a) MCP attached to the probe station. (b) Close up of the probe station.

Figure 4.3: Picture of the setup used for measurements in the clean room.

flowing across it, which is given by:

$$i_C(t) = \frac{dQ(t)}{dt} = C(t)\frac{dV(t)}{dt} + V(t)\frac{dC(t)}{dt} \ , \tag{4.4}$$

where $V(t)$ represents the voltage drop existing between rotor and stator.

Considering the voltage $V(t)$ as a combination of a continuous bias voltage V_O and an alternate voltage $v_p(t)$ (which is the sinusoidal high-frequency probe-signal), one obtains:

$$i_C(t) = C(t)\frac{dv_p(t)}{dt} + (V_O + v_p(t))\frac{dC(t)}{dt} \ . \tag{4.5}$$

Thus, since $v_p(t)$ has a much higher frequency than the capacitance $C(t)$ frequency, then $\frac{dv_p(t)}{dt} \gg \frac{dC(t)}{dt}$ and the following approximation holds:

$$i_C(t) \approx C(t)\frac{dv_p(t)}{dt} \ . \tag{4.6}$$

Therefore it is possible to detect the relative motion between rotor and stator by evaluating the current $i_C(t)$ as it is proportional to $C(t)$, (which in its turn, as already mentioned, is proportional to the rotor displacement). In practical terms, making reference to Fig.4.4, an actuation voltage is provided to stator A, a signal for current modulation is applied to the rotor and an amplifier, which reads the current, is attached to stator B, that is consequently polarized

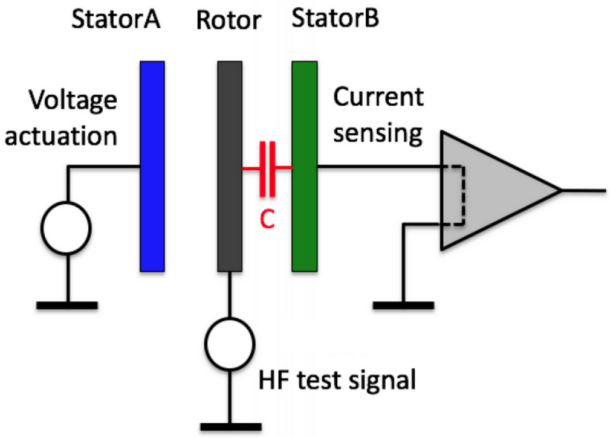


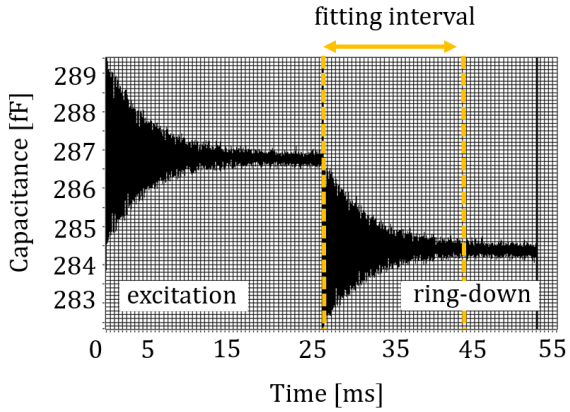
Figure 4.4: Scheme of the functioning of the MCP.

to the correct voltage [23].

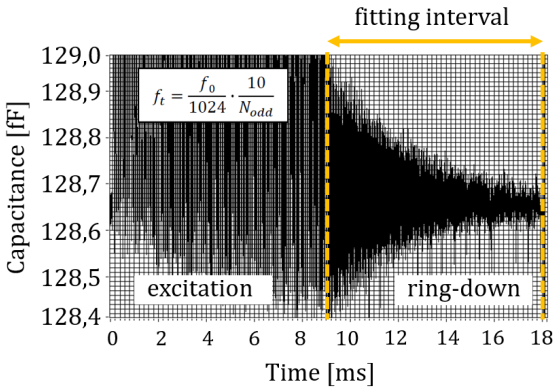
In a second moment of the experimental campaign, due to problems in the measurements (explained in details in the following), the test dies have been glued and wire bonded on LCC68 ceramic carriers, which can be mounted on a plastic socket that directly interfaces with the MCP and the waveform generator. Therefore pads are no more contacted with the probe station and the second campaign of measurements has been carried out outside the clean room.

4.4 First campaign of measurements: probe station

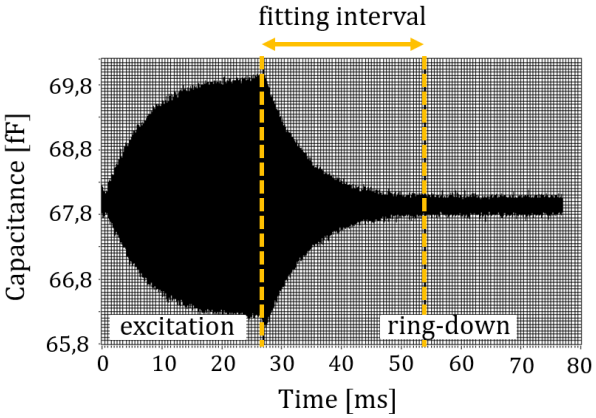
Each package contains the eight structures shown in Fig.4.1. The experimental campaigns are carried out on 11 packages, with an amount of 88 chips tested. For sake of simplicity the 11 packages are named progressively with alphabetical letters (therefore from A to M), while the eight structures in each package are named as reported in Tab.4.2.



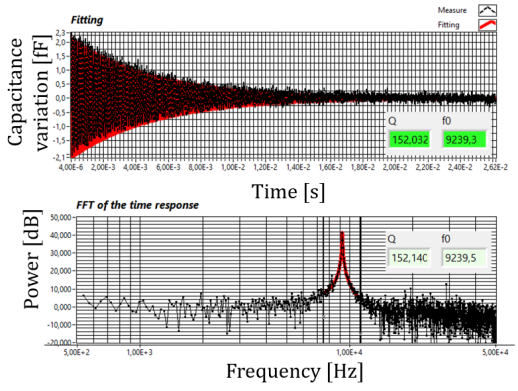
(a) Square wave excitation and consequent ring-down response.



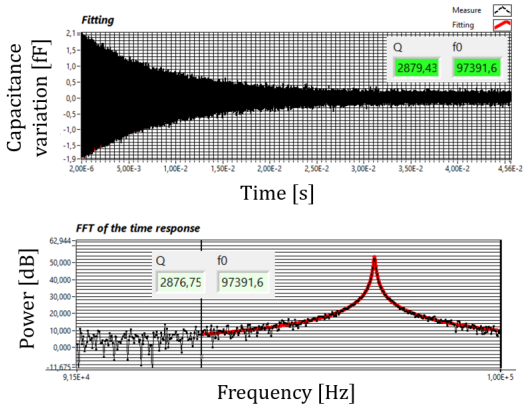
(b) Pulse-train excitation and consequent ring-down response.



(c) Burst signal excitation and consequent ring-down response.



(d) Example of non-accurate fitting, obtained after the square wave excitation.



(e) Example of a very good fitting, obtained after short-burst signal excitation.

Figure 4.5: Summary of the three adopted measurements approaches.

Table 4.2: Identifier codes of the eight devices in a package.

	frequency			
	10 kHz	**20 kHz**	**50 kHz**	**100 kHz**
gap = 1.8 μm	S4	S3	S2	S1
gap = 1.2 μm	S8	S7	S6	S5

4.4.1 Step approach

The first technique used to experimentally measure the Q factor of the test devices consists in the application of a 4 V slow square wave, shown in Fig.4.5(a). The square wave excites all the tilting masses in the package, then the ring-down response is fitted in order to estimate the quality factor. The fitting is performed both in time and in frequency (Fig.4.5(d)) with the aim of verifying that the two fittings agree one another.

However, since all the eight structures feature a single electrode for the drive and rotor pads, this complicates the actuation of the structures at high frequency. In other words, the structure with the lowest pull-in voltage (about 6-7 V) limits the actuation with ring-down methods to about 4 V for all structures. As a consequence, devices with higher resonance frequencies require a too long acquisition time in order to have a good Signal-to-Noise ratio (SNR).

4.4.2 Pulse-train approach

In order to speed up the simulation and to avoid the simultaneously excitation of all the structures in a package, the pulse-train technique has been adopted.

A 4 V slow square wave is supplied as well to the structures, but the resulting spectrum is fitted on the expected peak, just to estimate the nominal resonance frequency of the MEMS of interest.

Then a 3.5 V train-pulse (Fig.4.5(b)) is applied at an odd divider of the found resonance f_0, according to the following relation:

$$f_t = \frac{f_0}{1024} \cdot \frac{10}{N_{\text{odd}}} \ , \tag{4.7}$$

where 1024 at the denominator (equivalent to 10 bit) are the levels for the train codification and 10, at the numerator, is a rescaling factor to adapt to the component which generates the pulse-train. In this way the odd harmonics of the pulse-train will only excite the resonance frequency of interest. In the end, the spectrum is fitted both in time and in frequency again to extract resonance and Q factor.

However problems arise also with this approach because time fitting sometimes is still not reliable due to cross-talk. Therefore another technique is needed to carry out measurements.

4.4.3 Short-burst signal approach

In order to avoid issues encountered with the previous adopted techniques, an external waveform generator is introduced in the setup. The waveform generator is used to provide a short-burst signal close to the resonance frequency of the structure of interest with the following advantages:

- only a device among the eight in the package is excited at its own resonance. This enables to lower the applied voltage while holding a good SNR for the considered structure (thanks to Q factor amplification), without perturbing the other structures;
- the method provides the best signal amplification, as shown in Fig.4.5(c);
- the result is an optimum fitting, both in time and in the frequency domains, as depicted in Fig.4.5(e).

Obviously, to be sure about the equivalence of the three excitation approaches for Q measurements described so far, a cross-check on a subset of devices is performed by measuring the Q with all the described techniques. The obtained results confirm the equivalence of the three techniques, within reasonable deviations.

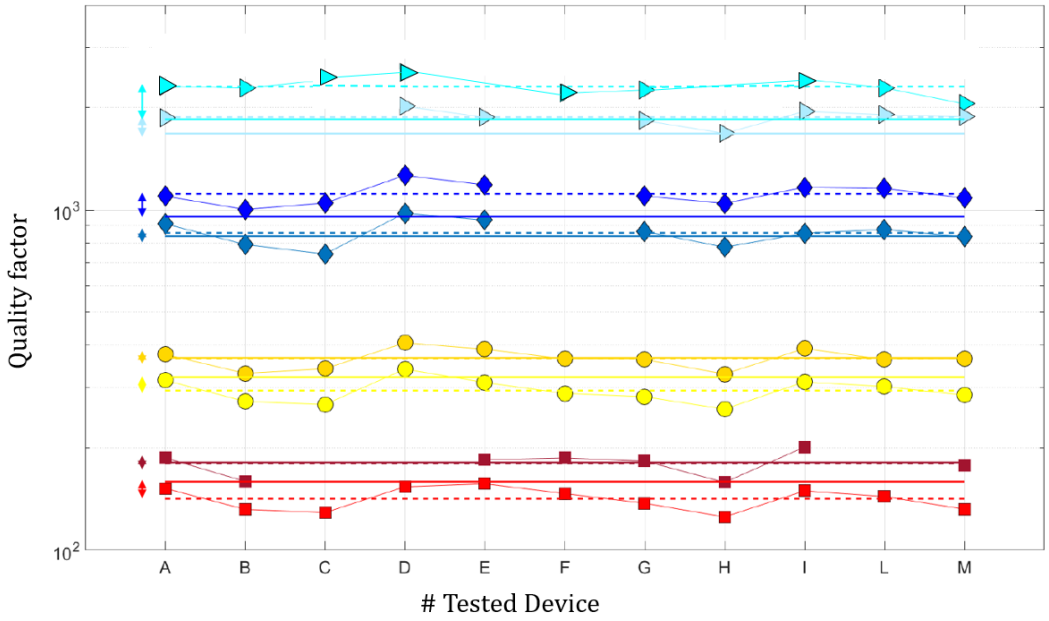


Figure 4.6: Experimental results using the probe station. Red → 10 kHz, yellow → 20 kHz, blue → 50 kHz and cyan → 100 kHz. Besides, in each color group: the highest curve with markers → 1.8 μm gap; the lowest curve with markers → 1.2 μm gap.

4.4.4 First campaign results

In Fig.4.6 is depicted a summary of the experimental results obtained with the probe station.

The $x-$axis reports the tested die, identified by a letter as already mentioned, while the $y-$axis reports the measured quality factor Q.

Each group of colors corresponds to a frequency range of the test structures, according to the following color codes:

- red → 10 kHz;
- yellow → 20 kHz;
- blue → 50 kHz;
- cyan → 100 kHz.

Within each color group the following distinction holds:

- the lowest curve with markers corresponds to measured data at 1.2 μm vertical gap;
- the highest curve with markers corresponds to measured data at 1.8 μm vertical gap;
- dashed lines are the corresponding measured average Q;
- solid lines are the theoretical predictions from the old numerical model at 0.7 mbar.

Low frequency devices (10, 20 and 50 kHz) present a good matching between measured quality factors and the predicted values at low frequencies, as expected. They have good repeatability in the measured Q, when mounting the devices in different configurations on the probe station. Moreover the quality factor Q for different dies presents a similar trend, which means that Q correctly follows the pressure in the package and varies correspondingly on different devices.

On the other hand, for high frequency devices (100 kHz), one can notice in the plot that several devices are missing because of too unreliable Q values, which vary between 1000 and 2500 in different measurements configurations.

Indeed it is observed that the Q factor significantly changes, for 100 kHz devices, depending on how the die is mounted for testing at the probe station: either fixed with adhesive tape, or fixed with alternative techniques. Moreover, for the two former cases, if the die is removed and put again on the probe, the quality factor may change unpredictably.

However, once a mounting is done, the chosen measurement technique has no impact on measured Q, thus changes above are independent of the characterization technique.

A possible explanation of the observed effect which generates differences depending on the mounting conditions is that the system can not be modelled as a single-degree-of-freedom spring-mass-damper system, since the coupling to the substrate is not rigid enough. Therefore the system can be seen in series with another spring-damper that connects the package to the probe. Depending on how the mounting is done, this new spring and damper may

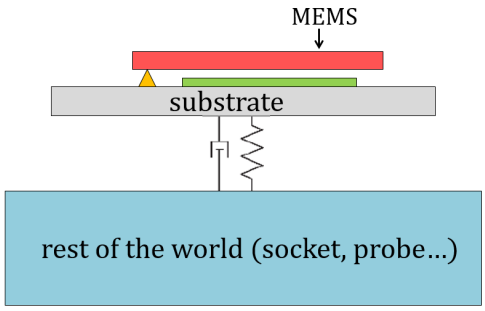


Figure 4.7: Graphical explanation of the effect causing differences in measurements depending on the mounting.

change their values, changing the way energy is transmitted and lost through and towards the substrate. Thus the measured Q is that of the entire two-degrees-of-freedom system and no longer that of the designed MEMS alone. See Fig.4.7 to better understand.

Hence the need to conduct a second experimental campaign, outside the clean room, with structures mounted on a ceramic socket, wire bonded and further fixed with non-conductive glue, in order to make the coupling between the MEMS and the substrate sufficiently rigid.

4.5 Second campaign of measurements: socket

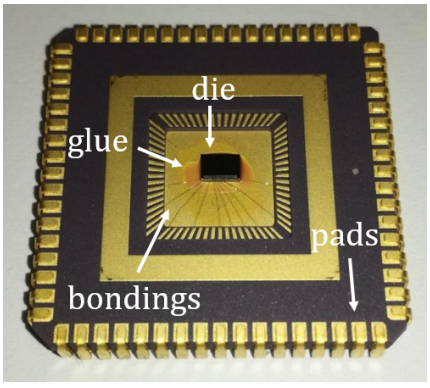


Figure 4.8: Ceramic carrier where the die is mounted. It is also connected to the pads by means of wire bondings.

In this second campaign, dies are mounted on sockets as shown in Fig.4.8.

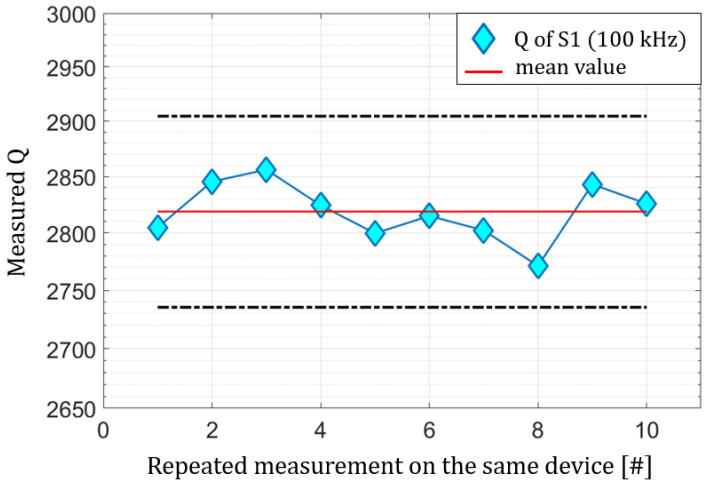


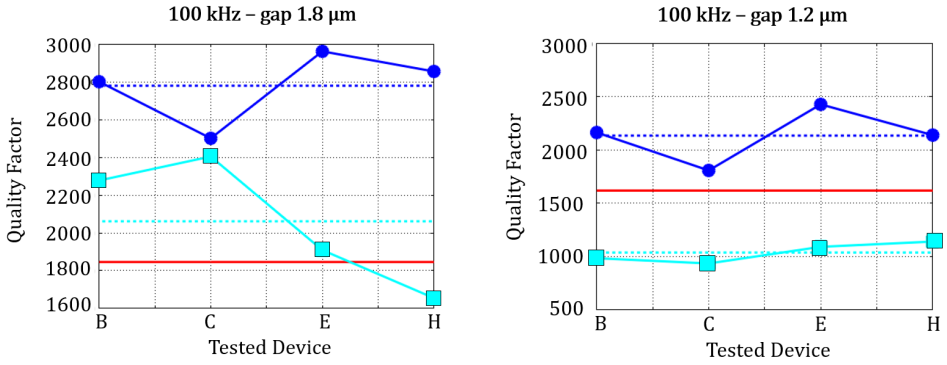
Figure 4.9: Results of the repeatability measurements on a sample with 1.8 μm vertical gap and 100 kHz resonant frequency.

Thanks to the bonding, actuation of devices and acquisition of signals are made easier and there is no need to access the devices in the package by means of the probe station. For this reason this second campaign is carried out outside the clean room.

The adopted measurements technique is the last presented, the short-burst signal approach, since it turned out, from the first campaign, to be the most accurate and satisfying.

4.5.1 Measurement repeatability

In order to verify that this alternative mounting of the devices provides more reliable results on Q factor extraction, contrary to the mounting on the probe station, a repeatability campaign is firstly performed on samples mounted on socket. For ten consecutive times, the MEMS glued on the carrier is disassembled from the holding socket, reassembled and finally re-measured. Results of one 100 kHz sample are reported in Fig.4.9, showing a repeatability in the quality factor estimation better than $\pm 1\%$. At lower frequencies, repeatability is even better as the larger achievable structure motion delivers higher SNRs, easing the fitting procedure.



(a) Comparison between the results of devices with 1.8 μm gap and 100 kHz resonance frequency.

(b) Comparison between the results of devices with 1.2 μm gap and 100 kHz resonance frequency.

Figure 4.10: Comparison between the results of the first campaign (cyan markers) and the second campaign (blue markers). The dashed lines are the average measured Q, while the red line is the theoretical value.

Since this new setup turned out to be more reliable than the first, in terms of Q factor prediction, new measurements on the devices are performed and their results are presented in the following.

4.5.2 Second campaign results

Among the tested dies in the first campaign, four of them (B, C, E and H) have been selected to be mounted on the socket, with the following logic: B and C have "well-functioning" MEMS vibrating at 100 kHz, in the sense that their measurements of Q, during the first campaign, are satisfying; E and H, on the contrary, are chosen because provide unreliable values of Q.

In Figg.4.10(a) and 4.10(b) is depicted the comparison between measurements carried out in the first campaign (solid curve with cyan markers) and measurements carried out in the second campaign (solid curve with blue markers) concerning 100 kHz devices with a vertical gap equal to 1.8 μm and 1.2 μm respectively. The solid red line, in both figures, represents the theoretical value of the quality factor, provided by the old numerical model. The dashed lines represent the average of the measured quality factors. In both cases, average Q is much higher than the theoretical quality factor. This clearly indicates the

presence of errors in the measurements conduct with the probe station.

These results, therefore, validate the formulated hypothesis in Sec.4.4.3: since dies are glued on socket, their coupling with the substrate is rigid enough to allow the measurement of the quality factor of the only MEMS in the package, which results to be higher than the Q measured during the first campaign.

Besides they validate the new proposed model, which predicts a more than linear growth of the quality factor with the resonance frequency, since measured Q are sensibly higher than theoretical Q (red line in the figure) predicted by the old model.

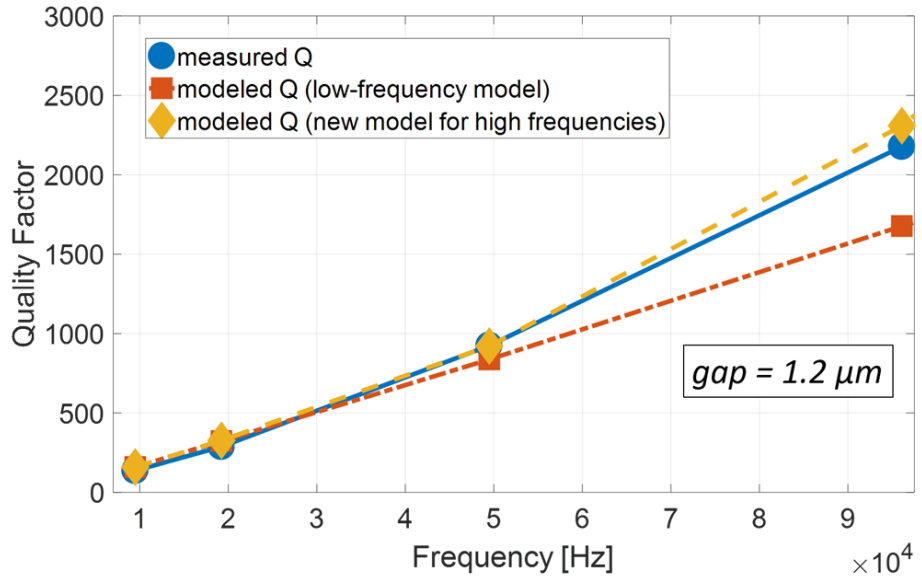
4.6 Summary

All the results obtained so far (theoretical, numerical and experimental) are now compared. Making reference to Figg.4.11(a) and 4.11(b), measured Q factors are reported on the $y-$axis as a function of the measured resonance frequency, which is reported on the $x-$axis.

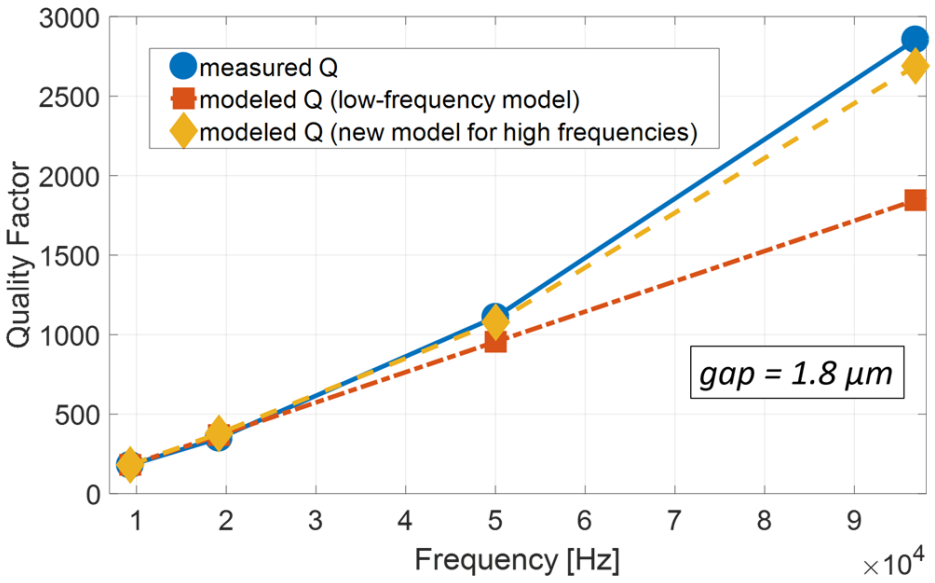
The dash-dotted curves shown with orange squares represent, in both figures, the numerical results obtained adopting the "quasi-static" prediction, therefore provided by the old model. Note how this foresees a constant damping coefficient with frequency, and thus a linearly growing Q factor. The dashed curves reported through yellow diamonds represent the numerical results obtained with the new model, from which the progressive divergence of this prediction with respect to the "quasi-static" one is evident as the frequency increases.

The experiments, here reported through solid curves and circle blue markers, confirm the model predictions: a deviation of the quality factor from the low-frequency linear dependence with resonance is effectively observed at 50 kHz, becoming clearly evident at 100 kHz. Such behaviour well matches the numerical results of the proposed tool, with $Q-$factor estimation errors lower than 8% at the largest analyzed frequency, for both the gap values.

Devices vibrating at even higher frequencies ($>$ 100 kHz) are currently under investigation. In this case, progressive decrease of the fluid damping contribution, as well as the corresponding increase of the thermoelastic damp-



(a) Results comparison of devices with 1.2 μm gap.



(b) Results comparison of devices with 1.8 μm gap.

Figure 4.11: Experimental results of Q factor measurements (circles) compared to low-frequency model (squares) and the new proposed model (diamonds), as a function of frequency.

ing, make the two contributions comparable.

The final simulation result sensibly depends on material parameters like the thermal expansion coefficient, the heat capacitance and the conductivity, which are not easily available for polysilicon in the literature and are, therefore, under investigation.

Conclusions

The work of thesis presented so far, proposes, implements and validates a new numerical model for the prediction of damping coefficient. Such a tool is of paramount usefulness in the design of MEMS sensors, since it allows an a priori estimate of the performances of such designed devices.

The work has been carried out into three phases: at first a research on existing models in literature and the analysis and comprehension of the proposed model has been made; then improvements in the code are implemented in order to make it efficient as much as possible; at last, experimental measurements and numerical simulations have been performed.

The analysis on the existing models provided information about their drawbacks and their limitations. Indeed, some of them hold only in particular physical conditions or for low resonance frequencies, while others, of wider validity, result to be computationally inefficient. In order to overcome limitations of the first category of existing models, the quasi-static hypothesis needed to be relaxed, as explained in Chapter 2, and the only hypothesis of small perturbations has been assumed in the formulation of the new model. Thus the model presented herein can be considered a generalization of the others, since it holds both for low and for high resonance frequencies, with a resulting range of frequencies that extends from 10 kHz to 100 kHz, which is expected to be of interest for next-generation MEMS inertial sensors.

On the other hand, our model needed to be improved in terms of computational speed, so as to be competitive with the second category of models which make use of the Test Particle Monte Carlo method. Hence the numerical implementation, which constitutes the main part of this work, has focused on the speeding up of the implemented model, as well as the overcome of some

errors found in the previous implementation.

The speeding up activity has only concerned the visibility part of the code, as explained in Chapters 2 and 3. Indeed, since MEMS devices have complicated geometries, they can be split into elementary units which are additionally partitioned into elementary quadrangular surfaces. Among all the surfaces, only those "visible" are taken into account for the computation of damping forces on the whole MEMS structure. This leads to the simulation of fully 3D structures by means of surface integral relations.

Numerical simulations on time performances provided successful results.

In the end, experimental measurements have been conducted in order to validate the proposed model. Their results confirm the reliability of the model with errors in the quality factor Q estimation lower than 8%.

Further studies reveal that, with increasing frequency (> 100 kHz), damping sensibly depends on material parameters, which are not easily available for polysilicon in the literature. Therefore the line of research, concerning the study of polysilicon properties, is still open. The aim is that of understanding if other sources of dissipation, like the thermoelastic damping, become comparable with the fluid damping at higher resonance frequencies. Such topic is of interest for the company which provides us the devices. Therefore devices vibrating at even higher frequencies are currently under consideration.

Nonetheless the model developed so far and presented in this work has proved to be fast, reliable, of wide application and provides very satisfactory results.

Bibliography

[1] A. Corigliano, B. D. Masi, A. Frangi, C. Comi, A. Villa, and M. Marchi, «Mechanical characterization of polysilicon through on-chip tensile tests», *Journal of Microelectromechanical Systems*, vol. 13, no. 2, pp. 200–219, 2004, ISSN: 1057-7157. DOI: 10.1109/JMEMS.2003.823221.

[2] B. Kim, M. A. Hopcroft, R. N. Candler, C. M. Jha, M. Agarwal, R. Melamud, S. A. Chandorkar, G. Yama, and T. W. Kenny, «Temperature Dependence of Quality Factor in MEMS Resonators», *Journal of Microelectromechanical Systems*, vol. 17, no. 3, pp. 755–766, 2008, ISSN: 1057-7157.

[3] C. Zener, «Internal Friction in Solids II. General Theory of Thermoelastic Internal Friction», *Phys. Rev.*, vol. 53, pp. 90–99, 1 1938. DOI: 10.1103/PhysRev.53.90. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRev.53.90.

[4] V. Kempe, *Inertial MEMS: Principles and Practice*. Cambridge University Press, 2011, ch. 3.

[5] T. Veijola, H. Kuisma, J. Lahdenperä, and T. Ryhänen, «Equivalent-circuit model of the squeezed gas film in a silicon accelerometer», *Sensors and Actuators A: Physical*, vol. 48, no. 3, pp. 239–248, 1995.

[6] C. Acar and A. Shkel, *MEMS Vibratory Gyroscopes: Structural Approaches to Improve Robustness (MEMS Reference Shelf)*, 2nd. Springer Publishing Company, Incorporated, 2008, ch. 4.

[7] M. Bao and H. Yang, «Squeeze film air damping in MEMS», *Sensors and Actuators A: Physical*, vol. 136, no. 1, pp. 3 –27, 2007, 25th Anniversary of Sensors and Actuators A: Physical, ISSN: 0924-4247. DOI: https://doi.org/10.1016/j.sna.2007.01.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0924424707000118.

[8] R. Christian, «The theory of oscillating-vane vacuum gauges», *Vacuum*, vol. 16, no. 4, pp. 175 –178, 1966, ISSN: 0042-207X. DOI: https://doi.org/10.1016/0042-207X(66)91162-6. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0042207X66911626.

[9] G. Bird, «Direct simulation and the Boltzmann equation», *The Physics of Fluids*, vol. 13, no. 11, pp. 2676–2681, 1970.

[10] A. Frangi, «A BEM technique for free-molecule flows in high frequency MEMS resonators», *Engineering Analysis with Boundary Elements*, vol. 33, no. 4, pp. 493 –498, 2009.

[11] S. Chapman, T. G. Cowling, and D. Burnett, *The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases*. Cambridge university press, 1970.

[12] C. Cercignani, «The Boltzmann equation», in *The Boltzmann Equation and Its Applications*, Springer, 1988, pp. 40–103.

[13] G. Bird, «Molecular gas dynamics and the direct simulation Monte Carlo of gas flows», *Clarendon, Oxford*, vol. 508, p. 128, 1994.

[14] A. Frangi, A. Ghisi, and L. Coronato, «On a deterministic approach for the evaluation of gas damping in inertial MEMS in the free-molecule regime», *Sensors and Actuators A: Physical*, vol. 149, no. 1, pp. 21 –28, 2009.

[15] M Abramovitz and I. Stegun, «Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables», *United States Department of Commerce, National Bureau of Standards*, 1970.

[16] P. Fedeli and A. Frangi, «Integral equations for free-molecule flow in MEMS: recent advancements», *Communications in Applied and Industrial Mathematics*, vol. 8, no. 1, pp. 67 –80, 2017.

[17] P. Fedeli, «Una tecnica BIE per la valutazione del gas damping di MEMS inerziali nel regime free-molecule», Master Thesis, Politecnico di Milano, 2013/2014. [Online]. Available: http://hdl.handle.net/10589/106942.

[18] B. Joe, «GEOMPACK — a software package for the generation of meshes using geometric algorithms», *Advances in Engineering Software and Workstations*, vol. 13, no. 5, pp. 325 –331, 1991, ISSN: 0961-3552. DOI: https://doi.org/10.1016/0961-3552(91)90036-4. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0961355291900364.

[19] https://people.sc.fsu.edu/~jburkardt/f77_src/geompack/geompack.html.

[20] A. Frangi, P. Fedeli, G. Laghi, G. Langfelder, and G. Gattere, «Near Vacuum Gas Damping in MEMS: Numerical Modeling and Experimental Validation», *Journal of Microelectromechanical Systems*, vol. 25, no. 5, pp. 890–899, 2016, ISSN: 1057-7157. DOI: 10.1109/JMEMS.2016.2584699.

[21] P. Fedeli, A. Frangi, G. Laghi, G. Langfelder, and G. Gattere, «Near Vacuum Gas Damping in MEMS: Simplified Modeling», *Journal of Microelectromechanical Systems*, vol. 26, no. 3, pp. 632–642, 2017, ISSN: 1057-7157. DOI: 10.1109/JMEMS.2017.2686650.

[22] A. Frangi and M. Cremonesi, «Semi-analytical and numerical estimates of anchor losses in bistable MEMS», *International Journal of Solids and Structures*, vol. 92-93, pp. 141 –148, 2016, ISSN: 0020-7683. DOI: https://doi.org/10.1016/j.ijsolstr.2016.02.038. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020768316001062.

[23] A. F. Longoni, *The ITmems-MCP (hand)book, A simple step-by-step guide to introduce the use of the MCP in MEMS characterization.* 2016.