Development of an Active

Damping Device

Quentin BOMBLED
Tutor : Dr. Olivier VERLINDEN

Postgraduate Advanced Degree

2005 - 2006

Departement of Theoretical Mechanics,
Dynamics and Vibrations
31, Boulevard Dolez
B-7000 MONS

Acknowledgements

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context

Nowadays, more and more applications refer to active vibration control or suppression. Vibrations are mostly undesired, for many reasons : perturbation to delicate devices, loss of comfort for people, deterioration of structures and maybe destruction of these ones. The energy being born from vibration can be dissipated either by passive or active devices. The passive damping is known for a long time, and had been extensively studied. It allows to decrease the level of vibration thanks to some physical modifications of the system (tuned mass), but one of the most important drawbacks is that the active bandwidth suppression is limited. Moreover, such absorbers are only effective if the excitation is nearly periodic. Finally, the addition of the absorber can also cause two resonant conditions to arise in place of the original single resonant condition [1]. These problems have been solved by adding an active element in parallel with the spring-damper system of the passive damping device [2]. In this work, we want to actively damp a rectangular plate by means of a loudspeaker.
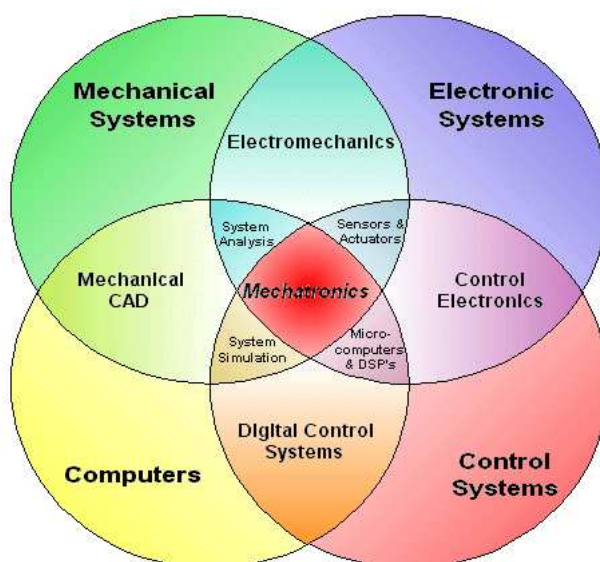
Figure 1.1: Definition of mechatronics

The active damping is considered like a mechatronic system, because of the high coupling between mechanics, electronics and data-processing (Fig.1.1). Mechatronics is defined in the following way : ”...combination of mechanical engineering (”*mecha*” for mechanisms),

electronic engineering (”*tronics*” for electronics), and software engineering. The purpose of this interdisciplinary engineering field is the study of automata from an engineering perspective and serves the purpose of controlling advanced hybrid systems” [3]. This work will be an introduction to this part of engineering.

1.2   Objective

The main aim of this work is to develop an active damping device, composed by a plate and a loudspeaker. In other words, the vibrations of the plate are going to be reduced thanks to the movement of the mass of the speaker. This system consists of a state feedback control, as represented in Fig.1.2. Because we want the damping of the plate, we will develop a regulator (command reference null, $r = 0$). The voltage applied to the actuator is simply the output of the plant multiplied by a gain, influencing performances of the system.
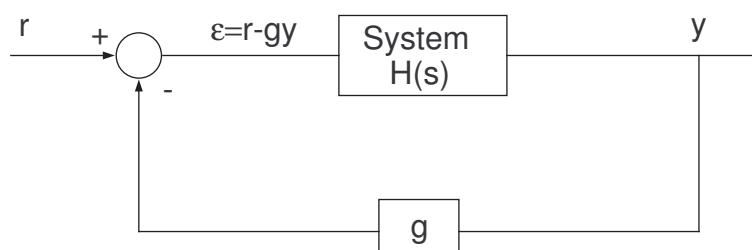
Figure 1.2: State feedback control

Two main aspects are identified in the following document :

- Equations of motion of the plate and the actuator, and the experimental identification of the parameters

- Control of the system thanks to a ”simple feedback”, and by means of state space feedback approach

As far as the state space approach is concerned, we first apply the theory to a simple system (inverted pendulum) in order to illustrate some notions like the observer or the *Kalman*'s filter. Finally, we will apply it to our problem.

1.3   State of the art

1.3.1   Actuators

The actuator is an essential element, conditioning a good working of the active damping system. Many parameters must be investigated, in order to make the good choice :

- mass : a shift of the natural frequencies can arise when the mass of the system increases. Heavier the structure is, less this effect will be marked on the studied system;

- force generated : an order of magnitude of the force acting in the system has to be known;

- command aspect : a command in current or in voltage has to be considered;

- cost : beside the price of the actuator, the whole measuring chain must be considered to estimate the cost of such a control.

(a) Representation of the loudspeaker   (b) Actuator glued on the plate
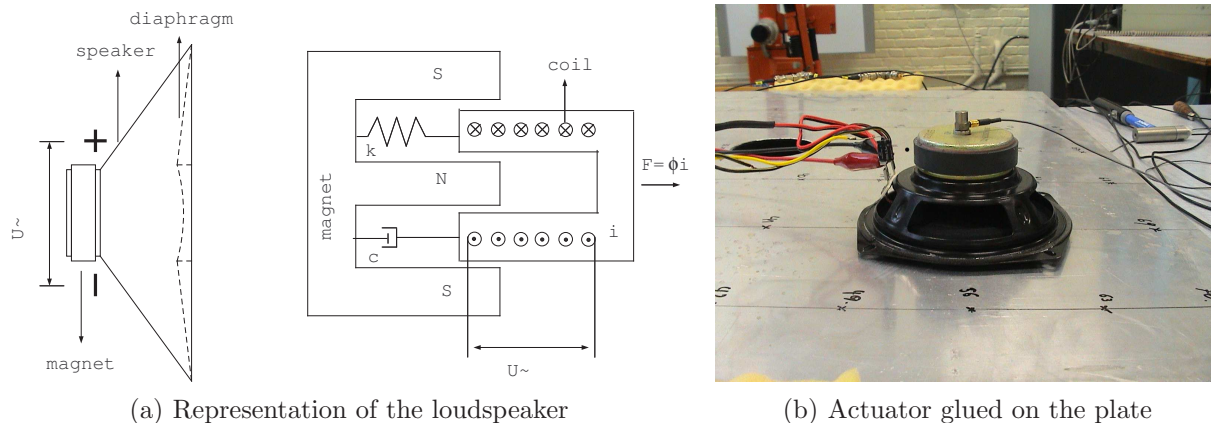
Figure 1.3: Proof-mass actuator

In section this section, the proof-mass actuators will be presented, which are well-adapted for our active damping problem. Of course, a lot of actuators exist nowadays, more and more efficient. The family of the actuators can be divided into three groups : electrical, mechanical and pneumatic and hydraulic actuators ([4], [5]). The first group is undoubtedly the most spread in mechatronic systems. The advantage of this kind of actuator is that the control is relatively simple and cheap, and the range of power covered is important, from nano-actuators to many kW of power are available. The last two categories are maybe more used for large structure control, or systems where the space requirement is not a dominating criterion.

Proof-mass actuator   A reaction mass is attached to a current magnetic coil moving in a magnetic field created by a permanent magnet (Fig.1.3(a)). The moving part is linked to the chassis thanks to a spring of stiffness k and a damper of constant c. A loudspeaker is a typical (and cheap!) proof-mass actuator. Ours will be described in section 2. In this work, the diaphragm is glued on the plate (Fig.1.3(b)), and the proof-mass is the magnet. The performance of such actuators are closely linked to the electromechanical constant ϕ, which gives the ratio between the force and the current going through the coil. Typical bode plots of this system can be found in Fig.1.4. We can see that, after the natural frequency of the actuator, this one behaves like an ideal force generator, with a large bandwidth and a no phase lag.
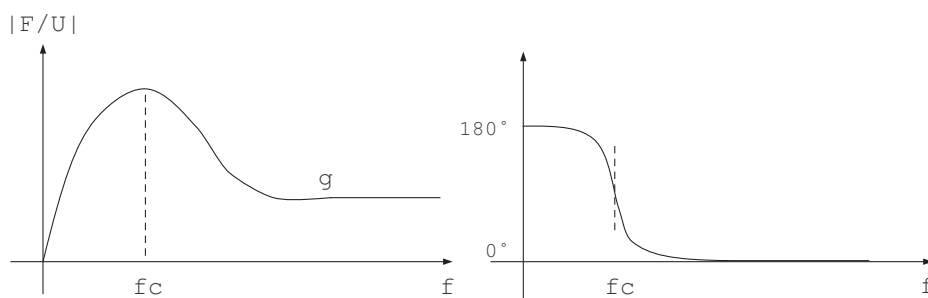
Figure 1.4: Bode plots of a proof-mass actuator

"Professional" proof-mass actuator can be found in [6] and [7]. Several criteria are taken into account for the conception, like avoiding a shift of natural frequencies of the structure (light weight), an important bandwidth and a high value for constant ϕ. The reference [8] gives informations about the design parameters of a lightweight proof-mass actuator, from an

electrodynamic and mechanical point of view.

As far as the ratio ϕ is concerned, if the current is sinusoidal, we have :

$$|F| = \phi i = m_{magnet}\ddot{x} = m_{magnet}\omega^2 d \tag{1.1}$$

where ω is the pulsation of the current, m the mass of the permanent magnet (approximately those of the speaker) and d the stroke of the actuator. We just mention here that a lot of problem can arise because of this stroke limitation, and that the performances and the control of this actuator can be degraded at low frequency (see [9] and [10]).

1.3.2   Collocated systems

By definition, a system is called collocated if the sensor and the actuator are at the same location (see Ref. [11]). Consider the simple case in Fig.1.5 :
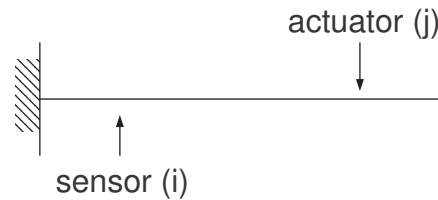
Figure 1.5: Cantilever beam with actuator (j) and sensor (i)

The transfer function between the excitation j and the response i can be written :

$$H_{ij}(\omega) = \sum_k \frac{B_{ijk}}{\omega_{0k}^2 - \omega^2 + j2\xi_k\omega_{0k}\omega} \tag{1.2}$$

where ω_{0k} is the natural pulsation of the structure, ξ_k is the damping ratio and B_{ijk} is the modal compliance, defined by $B_{ijk} = \psi_k^i\psi_k^j$ with ψ_k^i representing the modal amplitude at node i for the mode k. If we pose assume collocation, B_{ijk} will always be positive because $i = j$, and thus

$$\frac{dH_{ii}(\omega)}{d\omega} \geq 0 \tag{1.3}$$

Relation 1.3 shows us that the plot of the transfer function with respect to the frequency will lead to a monotonous growing function, which goes to $\pm\infty$ for the natural frequencies and is worth 0 for the anti-resonance. In all cases, there will be one and only one anti-resonance between two consecutive resonances, and consequently the system will be minimum phase due to pole-zero interlacing property (Fig.1.6 and 1.7).

We can emphasize here the advantage of collocation for active damping : the phase oscillates always between 0 and $-180°$, which gives a *minimum-phase* system. A brief overview about the research history in collocation system is given in [12], but we will retain only the fact that vibration control systems using collocated direct velocity feedback have an infinite gain margin, and so an unconditional stability. Let us remember that the instability is provoked by a gain $> 0dB$ and a phase lower than $-180°$. Thus, with a velocity feedback, the feedback gain can be very important without a destabilization of the close loop, because we never go under $-180°$ (see 1.7). The performances reached are thus higher in collocated system than in non-collocated ones.

A second notion must be introduced at this point, in order to complete this introduction : the duality sensor-actuator pair. *Duality* means energetically conjugated, like force and displacement

or velocity, or torque and angle. Sun in [13] shows us that there is not always unconditional stability for collocated sensors-actuators if they are not dual. Furthermore, he emphasizes that dual pairs are more robust if there is a de-collocation, which is very appreciable in case where perfect collocation is not possible.
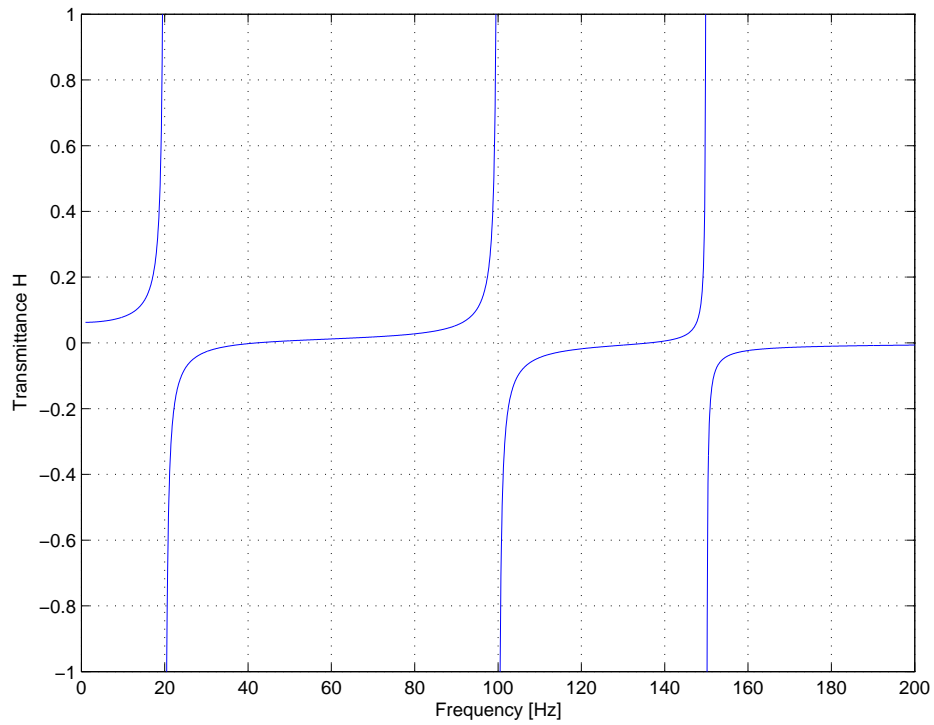
Figure 1.6: Alternating pole and zero
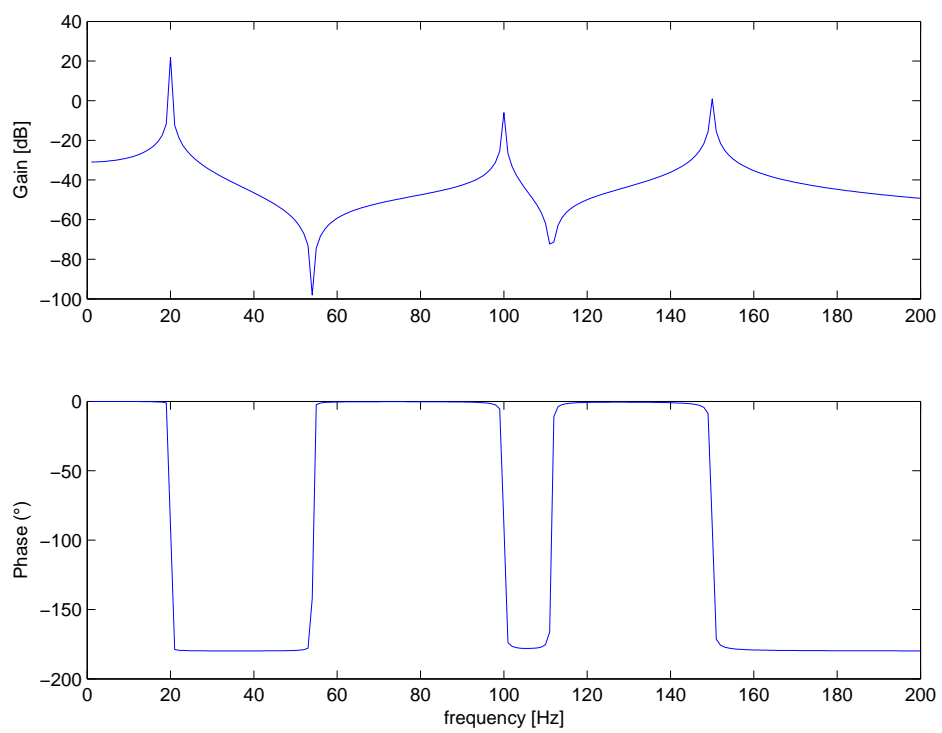
Figure 1.7: Pole-zero alternating behaviour

Chapter 2

Equations of motion
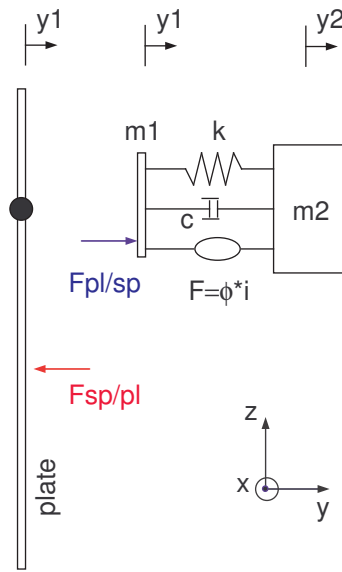
Figure 2.1: Modeling of the complete system

Fig.2.1 represents the modeling of Fig.1.3(b). The speaker is modeled by a two-mass system, from which mass m_1 is glued on the plate. m_2 represents the mass of the magnet. Equations of the plate and the loudspeaker are detailed below, while those of the coupled system are developed in section 2.3.

2.1   Plate

We are going to characterize the plate in the modal field, so as to have decoupled equations (Ref. [14]). If we suppose the equation of non-damped motion of the plate, we have :

$$[M]\ddot{\vec{x}} + [K]\vec{x} = \vec{F} \tag{2.1}$$

The conversion to the modal field is realized by introduction of $\vec{x} = [\phi]\vec{r}$ and by left multiplying 2.3 by $[\phi]^T$. The columns of the modal matrix $[\phi]$ corresponds to the mode shapes, while vector $\vec{r}$ gathers the modal amplitudes.

$$[M][\phi]\ddot{\vec{r}} + [K][\phi]\vec{r} = \vec{F} \tag{2.2}$$

$$[\phi]^T[M][\phi]\ddot{\vec{r}} + [\phi]^T[K][\phi]\vec{r} = [\phi]^T\vec{F} \tag{2.3}$$

$$m_i\ddot{r}_i + k_i r_i = \psi_i F_{sp/pl} \tag{2.4}$$

Finally we have a system of n decoupled equations, where subscript i refers to the considered mode. In chapter 4, we first develop the whole control for one mode, and extend it to several modes. With state-space approach in chapter 6, we will use two modes.

We assume here that the eigen vectors are normalized with respect to the modal mass, so $m_i = 1$ and $k_i = w_i^2$. The value of ψ_i corresponds to the modal deformation of the plate at the loudspeaker, for the considered natural frequency. $F_{sp/pl}$ represents the force applied by the loudspeaker on the plate, on the location where it is glued.

2.2   Loudspeaker

Fig.2.2 shows us the electromagnetical components of the loudspeaker. It is modeled by three elements in parallel : a spring, a damper (both composing the diaphragm) and an actuator.
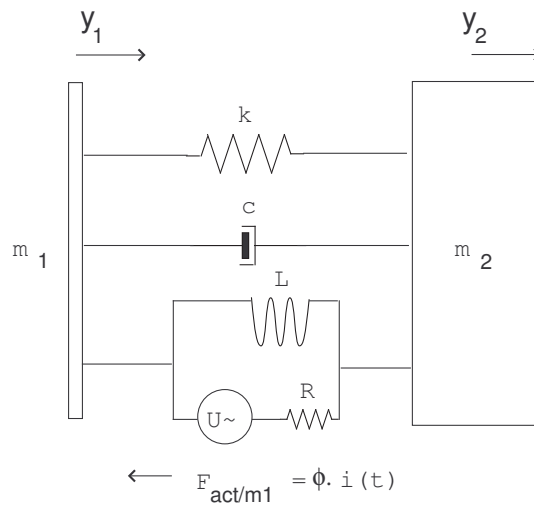
Figure 2.2: Modeling of the loudspeaker

We have two equations because of the two translational degrees of freedom y_1 and y_2 . The actuator of the speaker can be described by these equations :

$$U = E + Ri + L\frac{di}{dt} \tag{2.5}$$
$$E = \phi(\dot{y}_1 - \dot{y}_2) \tag{2.6}$$
$$F = \phi i \tag{2.7}$$

where U is the voltage applied on the loudspeaker, E the induced electrodynamic force, R and L the resistance and inductance of the loudspeaker respectively, ϕ is the electromechanical constant, and F the actuator force. If we neglect the influence of the inductance, which is very low (see section 3.2), we can rewrite the actuator force in terms of the force acting on m_1 : this is the force which will be applied on our plate:

$$F_{act/m_1} = \frac{\phi U}{R} - \frac{\phi^2(\dot{y}_1 - \dot{y}_2)}{R} \tag{2.8}$$

2.3   Coupled equations of motion

Following Fig.2.1, and considering only one mode, equation of the plate in the modal field is given by :

$$[M]\ddot{\vec{r}} + [K]\vec{r} = \vec{\psi}F_{sp/pl} \tag{2.9}$$

with $m_{ii} = 1$ and $k_{ii} = \omega^2$ because of the mass normalization of the eigen vectors. $\vec{\psi}$ is the vector gathering the deformed shapes. Equations of motion of mass m_1 and m_2 are (with m_1 neglected because the mass is very low):

$$F_{pl/sp} - k(y_1 - y_2) - c(\dot{y}_1 - \dot{y}_2) + \frac{\phi U}{R} - \frac{\phi^2(\dot{y}_1 - \dot{y}_2)}{R} = 0 \tag{2.10}$$

$$-k(y_2 - y_1) - c(\dot{y}_2 - \dot{y}_1) - (\frac{\phi U}{R} - \frac{\phi^2(\dot{y}_1 - \dot{y}_2)}{R}) - m_2\ddot{y}_2 = 0 \tag{2.11}$$

If one mode is considered, 2.9 and 2.10 give :

$$\ddot{r} + \omega^2 r = \psi(-k(y_1 - y_2) - c(\dot{y}_1 - \dot{y}_2) + \frac{\phi U}{R} - \frac{\phi^2(\dot{y}_1 - \dot{y}_2)}{R}) \tag{2.12}$$

Rearranging Eq. 2.12 and 2.11, and assuming we have $\dot{y}_1 = \sum_{i=1}^{n} \psi_i \dot{r}_i = \psi\dot{r}$:

$$\ddot{r} + \psi c(\psi\dot{r} - \dot{y}_2) + \frac{\psi\phi^2}{R}(\psi\dot{r} - \dot{y}_2) + \omega^2 r + \psi k(\psi r - y_2) = \frac{\psi\phi U}{R} \tag{2.13}$$

$$m_2\ddot{y}_2 + c(\dot{y}_2 - \psi r) - \frac{\psi\phi^2}{R}(\psi\dot{r} - \dot{y}_2) + k(y_2 - \psi r) = -\frac{\phi U}{R} \tag{2.14}$$

For clearer notation, m_2 and y_2 are simply called m and y in the continuation of the document. Next, we rearrange the system to obtain the two equations of the motion:

$$\ddot{r} + (\psi^2 c + \frac{\phi^2\psi^2}{R})\dot{r} + (\omega^2 + \psi^2 k)r - (\psi c + \frac{\psi\phi^2}{R})\dot{y} - \psi k y = \frac{\psi\phi U}{R} \tag{2.15}$$

$$\ddot{y} + \frac{c + \frac{\phi^2}{R}}{m}\dot{y} + \frac{k}{m}y - \frac{\psi c + \frac{\psi\phi^2}{R}}{m}\dot{r} - \frac{\psi k}{m}r = \frac{-\phi U}{mR} \tag{2.16}$$

We write the full system under the usual form $[M]\ddot{\vec{q}} + [C]\dot{\vec{q}} + [K]\vec{q} = [F]U$, with $\vec{q} = \{r\ y\}^T$:

$$[M] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad [C] = \begin{bmatrix} \psi^2 c + \frac{\phi^2\psi^2}{R} & -(\psi c + \frac{\psi\phi^2}{R}) \\ -\frac{\psi c + \frac{\psi\phi^2}{R}}{m} & \frac{c + \frac{\phi^2}{R}}{m} \end{bmatrix} \quad [K] = \begin{bmatrix} \omega^2 + \psi^2 k & -\psi k \\ \frac{-\psi k}{m} & \frac{k}{m} \end{bmatrix} \quad [F] = \begin{bmatrix} \frac{\psi\phi}{R} \\ \frac{-\phi}{mR} \end{bmatrix}$$

Chapter 3

Identification

This section presents the identification of the components composing our active damping device. It is an important part of the work, and it could be avoided if we used well-known components available on the market. But the identification is anyway interesting in its own.

3.1 Modal analysis of the plate

3.1.1 Physical properties and geometry
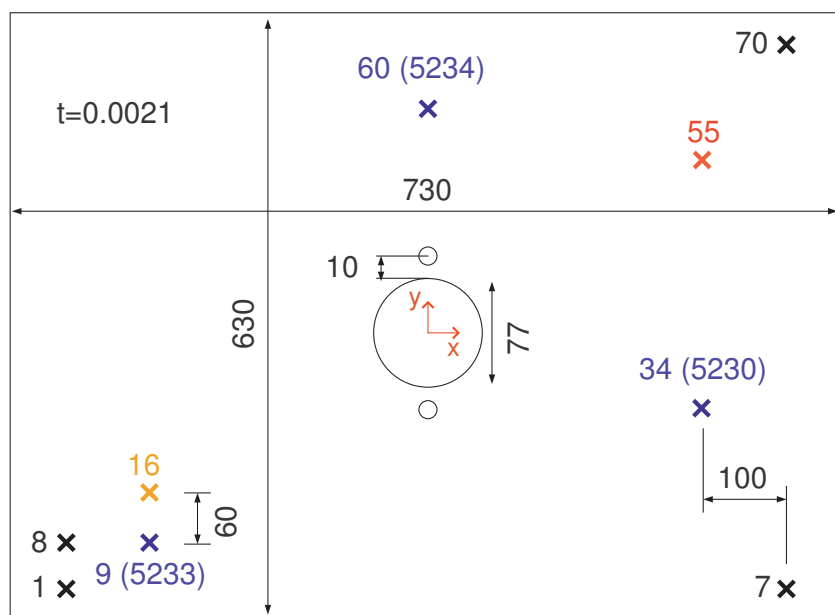
Figure 3.1: Instrumentation of the plate (sensor represented in blue)

The plate is in aluminum. Here are some mechanical properties :

- $E = 69\ GPa$

- $\nu = 0.33$

- $\rho = 2700\ kg/m^3$

Dimensions are given in Fig.3.1. The hole in the center of the plate has no special reason of existence. The two little ones, although they are represented, are not taken into account in our analysis.

17

- $L = 0.73\ m$

- $l = 0.63\ m$

- $t = 0.0021\ m$

- $\phi = 0.077\ m$

3.1.2   Experimental setup

Modal analysis is made by placing the plate on two little blocks of foam (see Fig.3.2), and without the loudspeaker. The plate is divided into 70 points. The numbering can be found in Fig.3.1. Three *Dytran 3100B* accelerometers are used; they all have a sensibility of $100mV/g$ (max. 3% of error). The weight of the accelerometers is about $0.067kg$. In experimental modal analysis, it is recommended that the weight of the sensor does not exceed the weight of the structure of 1%. Here we reach 7%. But we will see in paragraph 3.1.6 that experimental and numerical approach agree well, so we consider that the influence of the sensors is very weak in our modal analysis.
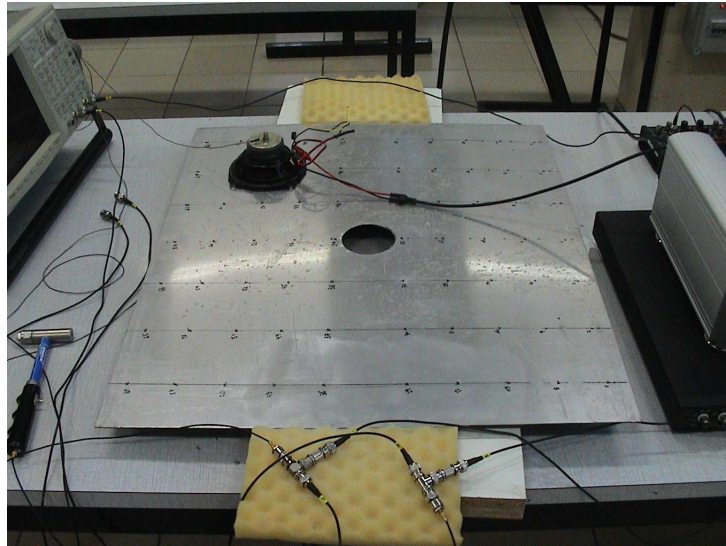
Figure 3.2: Plate placed on foam

The system is excited by a hammer. The force sensor incorporated inside this one is an `IEPE Dytran` 5850A with a sensibility of 44.48 N/V.

The modal analysis methods generally consider H_{ij} transmittances rather than H_{ji}[1]. Therefore, thanks to the reciprocity theorem of Betti-Maxwell ($H_{ij} = H_{ji}$), we can displace the hammer and stay fix the accelerometers, which is more convenient.

Parameters of the acquisition are summarized in Table 3.1. To perform the analysis, we use the `LMS Scadas` system.

---

[1]with i relative to the acceleration and j to the excitation

Sampling frequency	$400\ Hz$
Number of samples per measure	2048
Period of sampling	$5,12\ s$
Resolution	$0,19\ Hz$
Observable frequency range	$[0\ Hz; 200\ Hz]$
Input windows	Force-Exponential (Cut-off $1,22\%$)
Output windows	Exponential (Exponential decay 100%)
Average	3 acquisitions
Excitation	Hammer, mid-soft nozzle

Table 3.1: Parameters of acquisition

mode	f_{LMS} (Hz)	$\xi_{LMS}(\%)$	f_{Ansys} (Hz)	Deformed Shape
1	6	3.05	-	rigid body
2	10.0	1.82	14.69	-
3	21.1	3.33	19.87	-
4	31.6	1.41	28.87	first longitudinal bending
5	38.4	1.11	36.64	bending-torsion
6	45.1	1.43	40.80	bending-torsion
7	54.1	2.37	59.30	-
8	73.7	0.79	72.67	-
9	78.9	0.61	79.06	second longitudinal bending
10	93.2	0.97	91.85	-
11	114.7	0.35	115.05	third lateral bending
12	122.8	0.76	121.53	-
13	130.1	0.82	132.21	-
14	146.5	0.24	155.05	third longitudinal bending
15	158.4	0.70	165.92	-
16	164.9	0.56	170.58	-
17	174.7	1.05	171.30	-
18	193.0	0.35	191.89	-
19	194.4	0.28	199.94	-

Table 3.2: Natural frequencies

3.1.3   Results

Fig.3.3 shows the sum of all the frequency responses. We can identify different modes, summarized in table 3.2.
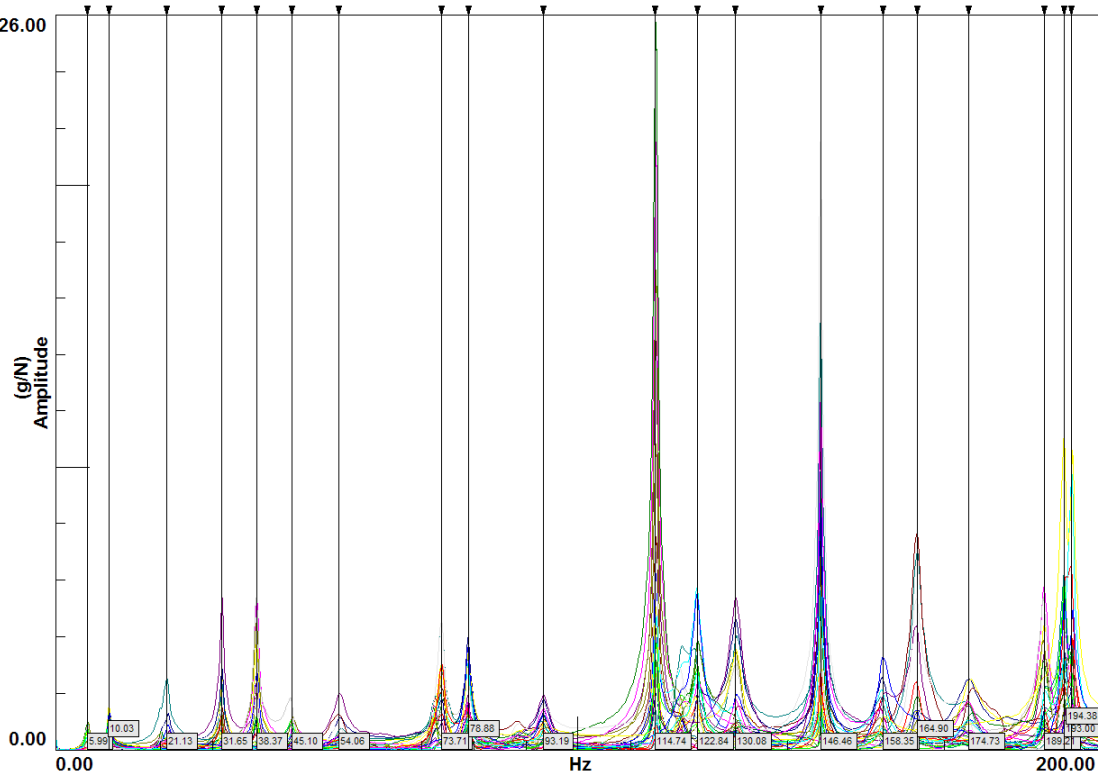
Figure 3.3: Sum of FRF

3.1.4   Finite element model

The natural frequencies can be determined thanks to experimental modal analysis (3.1.3). However, the mass and stiffness matrices needed in equation 2.1 of section 2 are not available. A finite element model allows us to access theses data. But an updating procedure on the mechanical parameters is recommanded to match the experimental and numerical frequencies. However, in our case, the chosen modes seems to be close enough to avoid it (see Table 3.2). The model of the plate is built from shell elements (SHELL63) and does not consider any damping.

3.1.5   Choice of the modes and location of the loudspeaker

In the first approach of active damping (simple feedback), all the modes can be theoretically damped (see section 4). On the other hand, the state space feedback approach needs to know what are the modes to damp. As far as the choice of modes is concerned, several criteria are defined :

- experimental and finite element model must agree. This is the case, we will speak about it in section 3.1.6;

- they must be far away from natural frequency of the loudspeaker (we develop that point in section 3.2), where the phase shift and the gain of its bode curves are considered as constant;

- in order to avoid the $50 Hz$ noise, we will choose higher natural frequencies than this value;

- it seems natural that active damping will have its best efficiency only if the loudspeaker is located at a maximum of vibration : in other words, the modes must have their maximum amplitude at the same node;

- number of modes is not infinite : in section 6.1.2, we will show that, in our case, only two modes can be damped (controllability limitation);

- the anti-aliasing filter in the measurement chain leads to an unavoidable phase lag (see section 6.3). In order to limit it, sampling frequency must be chosen between 20 to 100 times faster than observed phenomenas. So, it seems obvious that control high frequency leads to prohibited sample rate, and is very greedy in terms of computational resources;

Our choice goes to the two "higher" modes represented in Fig.3.5 (114 and $146Hz$). Frequency shifts between the two models are not important, and deformed shape agrees, but not perfectly. Let us remark that original plate is slightly folded, which can modify its stiffness. On the other hand, free-free assumption is not really respected, because the plate lies on some foam as detailed earlier.

Loudspeaker is glued on the plate at coordinates $(0.2, 0.1685, 0)$ with respect to considered axis of Fig.3.1 (node 55). This location represents a maximum of amplitude for the considered modes.
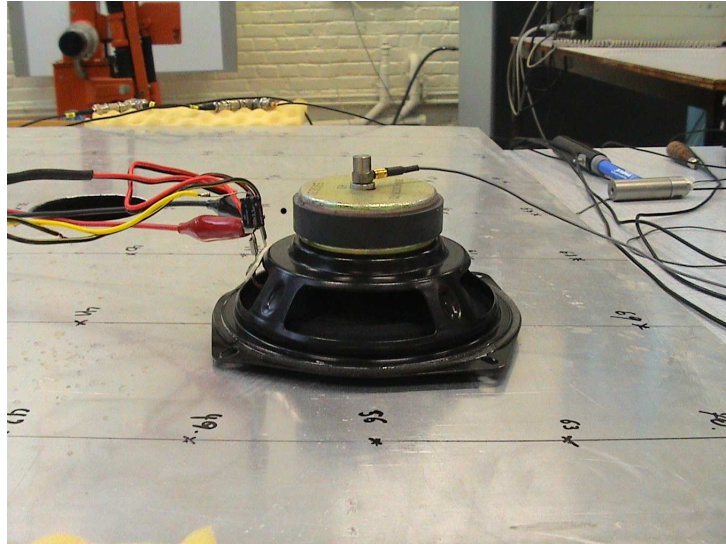
Figure 3.4: Loudspeaker located on node 55

3.1.6   Modal assurance criterion

This criterion allows us to verify the good agreement between the deformed shapes of the experimental and numerical analyses. The element MAC_{ij} (i^{th} row and j^{th} column) of the MAC matrix is given by

$$MAC_{ij} = \frac{\left[\left\{\vec{\psi_i}^{num}\right\}^T \cdot \left\{\vec{\psi_j}^{exp}\right\}\right]^2}{\left|\vec{\psi_i}^{num}\right|^2 \cdot \left|\vec{\psi_j}^{exp}\right|^2} \tag{3.1}$$

Table 3.3 shows us the agreement between experimental analysis and numerical simulation, for chosen modes : thus, we can expect a good accuracy by taking the mass and stiffness matrices in our model in section 2.
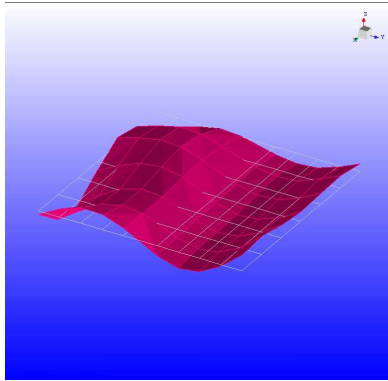
(a) Second longitudinal bending
mode (Ansys-78.8Hz)
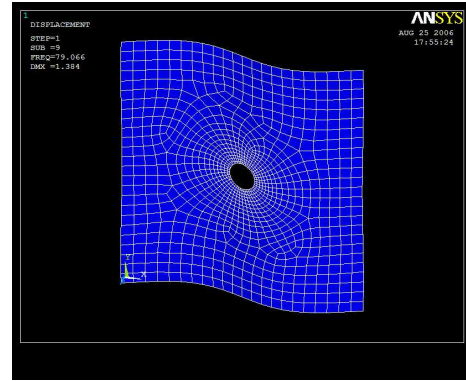
(b) Second longitudinal bending mode
(Lms-79.9Hz)
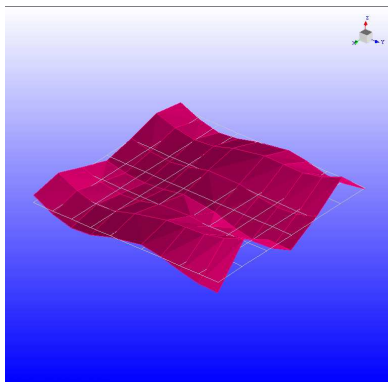
(c) Third lateral bending mode
(Ansys-114.7Hz)
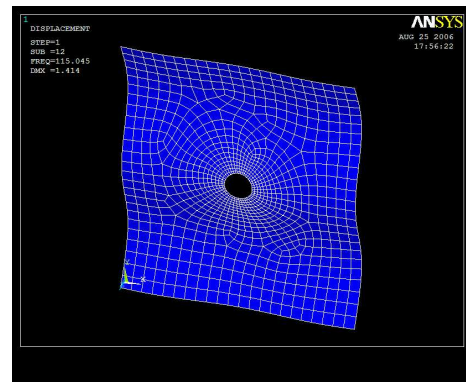
(d) Third lateral bending mode
(Lms-115.1Hz)
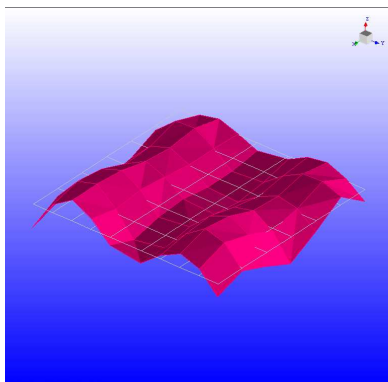
(e) Third longitudinal bending
mode (Ansys-146.5Hz)
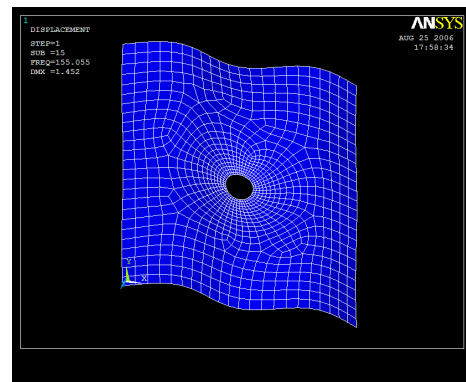
(f) Third longitudinal bending mode
(Lms-155.1Hz)

Figure 3.5: Experimental and analytical deformed shape

Mode	LMS	1	2	3
Ansys	Frequency	78.9 Hz	114.7 Hz	146.5 Hz
1	79.1 Hz	**67**	1	0
2	115.1 Hz	0	**62**	2
3	155.1 Hz	0	6	**64**

Table 3.3: MAC (%) for the chosen natural frequencies

3.2   Modal analysis of the loudspeaker

Our actuator is totally unknown : in order to compare numerical and experimental approaches, it is necessary to find its different mechanical and electrical characteristics. The model of the loudspeaker is represented in Fig.3.6
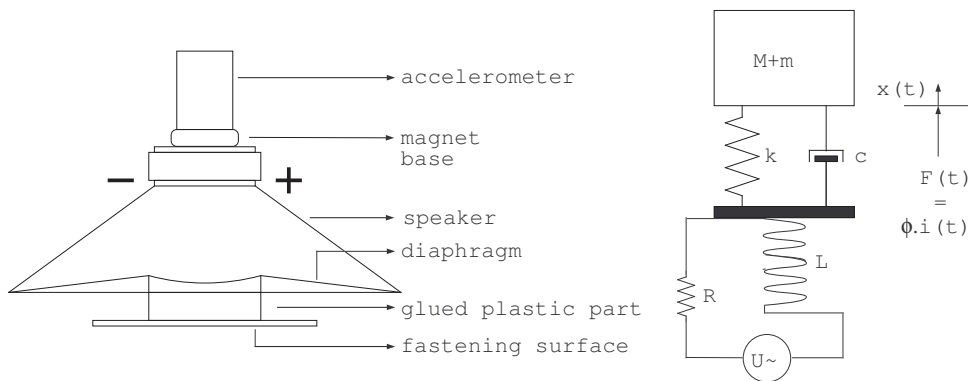
Figure 3.6: Identification of the speaker

The total mass $M + m$ (speaker and accelerometer) is excited by the vibration of the diaphragm when a sinusoïdal voltage is applied to the speaker. In this model, we have a coupling between an electrical circuit and a mechanical system. If we fix the fastening surface, here is the equation of the mechanical system :

$$(m + M)\ddot{x}(t) + c\dot{x}(t) + kx(t) = F(t) = \phi \cdot i(t) \tag{3.2}$$

where x is the displacement of the mass, c and k the damping and the stiffness coefficients, and F the actuator force. This one is coupled with the current in the electrical circuit, thanks to an *a priori* unknown constant ϕ. The behaviour of the electrical circuit is given by :

$$U(t) = \phi\dot{x}(t) + L\frac{di(t)}{dt} + Ri(t) \tag{3.3}$$

These equations can be compared to those of the DC motor. The voltage applied to the speaker is divided into a part counterbalancing the electromotive force ($\phi\dot{x}$), and a "useful" voltage which can activate the speaker. It is due to the Faraday's law : *"The inducted current tends to oppose itself to the cause which gave its existence"*. It explains why an increasing velocity lowers the mechanical force. ($E = \phi\dot{x}$ and $F = \phi i$).

If we use the Laplace's transform, and for initial conditions null, equations 3.2 and 3.3 become :

$$((M + m)s^2 + cs + k)X(s) = \phi I(s) \tag{3.4}$$

$$U(s) = s\phi X(s) + (R + sL)I(p) \tag{3.5}$$

From this relation, we can find

$$I(s) = \frac{U - s\phi X(s)}{R + pL} = \frac{U(s)}{R + sL} - \frac{s\phi X(s)}{R + sL} \tag{3.6}$$

Finally, if the inductance of the speaker is neglected (because L is very low, see section 3.2.1), the transfer function between the displacement and the voltage applied to the speaker is:

$$\frac{X(s)}{U(s)} = \frac{\frac{\phi}{R}}{((M+m)s^2 + (c + \frac{\phi^2}{R})s + k)} = \frac{\frac{\phi}{R(M+m)}}{(s^2 + 2\xi\omega_0 s + \omega_0^2)} \tag{3.7}$$

3.2.1 Informations available

M is the mass of the speaker and m is the one of the accelerometer. R and L can be estimated thanks to a multimeter and a selfmeter. The stiffness k, the damping c and the constant ϕ are going to be estimated thanks to the experimental FRF.

So, we have :

- $M = 400 \ gr$ (speaker)

- $m = 67 \ gr$ (accelerometer)

- $R = 5 \ \Omega$

- $L = 90 \ \mu H$

- $\tau_e = L/R = 18e - 6 \ s$

We see that the electrical time constant is not important in our problem : it gives a pole over $5 \ kHz$.

Acquisition

The following material has been used for the measurement of the FRF :

- Accelerometer : *DYTRAN* 3100B ICP, sensibility of 100 mV/g (S/N 5234)

- *SCADAS*, type SC305, 16 bits

- Power Amplifier *VELLEMAN* K8040

The acquisition principle is represented in Fig.3.7.
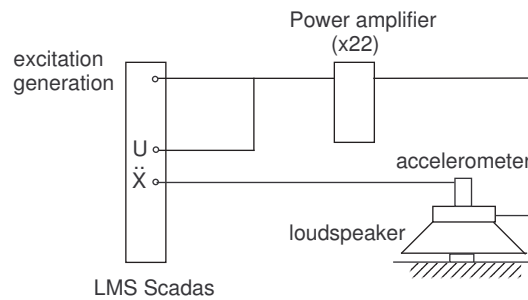
Figure 3.7: Acquisition setup

Characteristics of the acquisition are summarized in table 3.4

1st, 3rd and 5th acquisition	
Bandwidth	[0 Hz; 320 Hz]
Resolution	0.15 Hz (2048 spectral lines)
Windows	Uniform
Average on	3 acquisitions
Estimator	H1
Excitation	Level : 0.1 V Log (1) and Lin (3-5) Periodic chirp [0.001 → 120 %] of the acquisition bandwith
Added mass	0.067 kg (2,4) and 0.167 kg (6)
2nd, 4th and 6th acquisition	
Bandwidth	[0 Hz; 640 Hz]
Resolution	0.15 Hz (4096 spectral lines)
Windows	Uniform
Average on	3 acquisitions
Estimator	H1
Excitation	Level : 0.1 V Log (2) and Lin (4-6) Periodic chirp [0.001 → 120 %] of the acquisition bandwith
Added mass	0.067 kg (2,4) and 0.167 kg (6)

Table 3.4: Parameters of the acquisition

3.2.2   Analysis

In Fig.3.8, we represent the FRF ($\frac{\ddot{X}(f)}{U(f)}$) of the system for different acquisitions. The resonance frequency can be taken at $f_0 = 15.87$ Hz (so the stiffness k is about 4720 N/m), the peak being about 8.5 g/V (G_{max}). To have an estimation of the reduced damping coefficient, we may use the *peak picking* method. On Fig.3.8, the frequency band at -3 dB corresponds to those at $G_{max}/\sqrt{2}$. Thus :

$$\Delta f|_{\frac{G_{max}}{\sqrt{2}}} \simeq 2\xi f_0 \tag{3.8}$$

So we have a mean of $\xi = 0.08$.

The last parameter, ϕ, which represents the relation between the current and the force applied to the speaker, can be determined. We can write (if we take the infinite at 320 Hz and after):

$$\lim_{f \to \infty} \frac{\ddot{X}(f)}{U(f)} \simeq \lim_{f \to \infty} \frac{s^2 \frac{\phi}{(M+m)R}}{s^2 + 2\xi\omega_0 s + \omega_0^2} \simeq \lim_{s \to \infty} \frac{s^2 \frac{\phi}{(M+m)R}}{s^2 + 2\xi\omega_0 s + \omega_0^2} \tag{3.9}$$

And so

$$\frac{\phi}{(M+m)R} \simeq 1.2 \ g/V \simeq \frac{\phi}{0.467 kg \cdot 5\Omega} \tag{3.10}$$

Equation 3.10 gives us $\phi = 28$ N/A. It must be said that this value is computed by the *SCADAS* for an input value of 0.1 V. In reality, the input signal of the speaker comes from an amplifier ($\times 22!$) and so the value obtained must be divided by this ratio. We have thus $\phi = 1.4$ N/A.
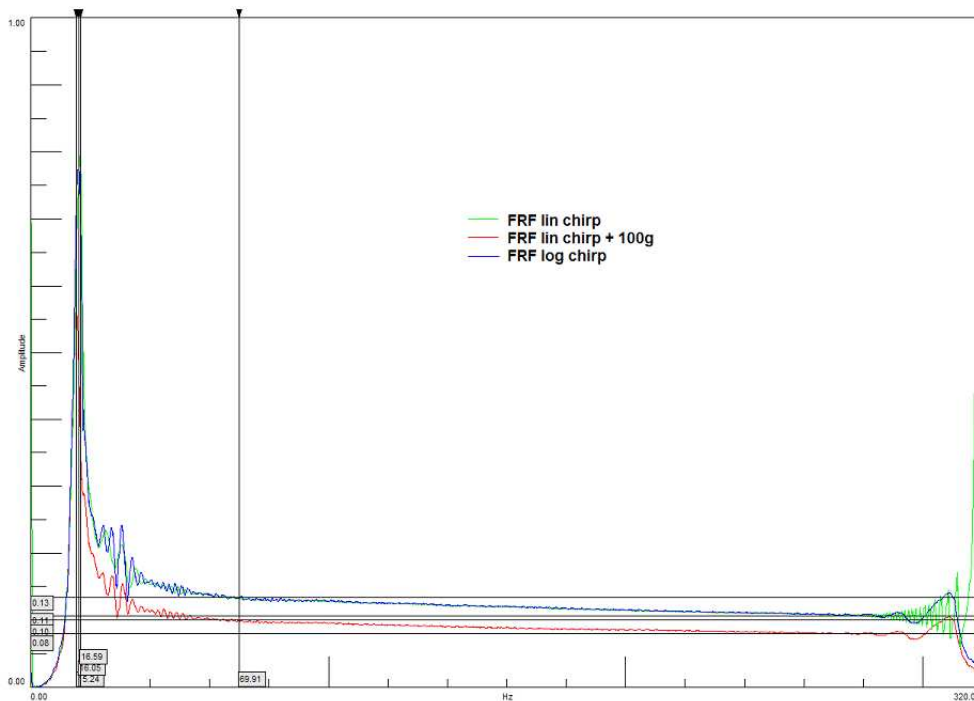
Figure 3.8: Frequency response function of the speaker (Acquisition bandwidth : 320 Hz)

3.2.3 And finally...

Table 3.5 resumes the characteristics of the speaker.

Electrical characteristics	
Resistance R	5 Ω
Inductance L	90 μH
Constant ϕ	1.4 N/A
Mechanical characteristics	
Mass M	0.400 kg
Mass (M+m)	0.467 kg
Natural frequency f_0 (ω_0)	16 Hz (100.5 rad/s)
Damping ratio ξ	8 %
Damping c	7.5 Ns/m
Stiffness k	4720 N/m

Table 3.5: Characteristic of the speaker

3.3   Conclusion

An experimental modal analysis and a complementary finite element modal have been used to identify the modes of the plate. In section 3.1.5, we give several argument concerning the choice of the modes, as well as the location of the actuator.

As said in the paragraph 1.3.1, the loudspeaker can be considered like as an ideal generator force, without phase lag, assuming that we are far enough of the mechanical natural frequency. It is a very low-cost actuator, and certainly powerful enough to achieve the development of an active damping device.

To conclude this section, it is interesting to note the that the performances are not driven by the mass ratio between the structure and the actuator such as with a tuned mass damper ([6]). Impressive damping ratio has been achieved by *Micromega Dynamics* on a $400kg$ structure with a $20gr$ actuator.

Chapter 4

Damping in SISO case : root locus approach

4.1 Theory

This section is a simplified approach of the control loop, allowing to understand more physically the concept of active damping before entering in more complex theory (state space feedback). We assume that the output of the plant is the velocity of the plate. We have one speaker on it, and we create the active damping on one mode (the third longitudinal bending mode at $\pm 155\ Hz$). We can make a feedback loop on velocity, multiply it by a gain and so we have an opposite velocity of the loudspeaker (see Fig.1.2). The efficiency of the damping is depending on the gain feedback, and the location of the loudspeaker. If, for the considered mode, the loudspeaker is at a vibration node, it is evident that the active damping will be inefficient. The location of the loudspeaker has been considered in paragraph 3.1.5.

For a simple velocity feedback, the voltage command of the system becomes

$$U = -g\dot{y}_1 = -g\psi\dot{r} \tag{4.1}$$

Combining Eq. 4.1, 2.15 and 2.16, we find a new damping matrix C, expressed in the following way :

$$[C] = \begin{bmatrix} \psi^2 c + \frac{\phi^2\psi^2}{R} + \frac{\psi^2\phi g}{R} & -(\psi c + \frac{\psi\phi^2}{R}) \\ -\frac{\psi c + \frac{\psi\phi^2}{R}}{m} + \frac{\psi\phi g}{mR} & \frac{c + \frac{\phi^2}{R}}{m} \end{bmatrix}$$

Now,the representation of the root locus allows us to see the performances of the loop by varying the gain g (Fig.4.2). Remember that the root locus is the evolution of the system poles with respect to the variation of the feedback gain, and that the curve are beginning at pole in open loop ($g = 0$) and are going to the zeros in open loop ($g = \infty$). We recall in Fig.4.1 how to find the damping ratio and the frequency of the pole.

The big loop of Fig.4.2 represents the evolution of the pole for the plate, and the small one for the loudspeaker. We see clearly that the system is not always stable. Beyond a critical value, the system becomes unstable because of the loop concerning the loudspeaker. Here, with a feedback gain of 116, we reach a damping ratio of 1.5%, while without closing the loop, the damping ratio is only due to the loudspeaker and is about 0.5% (no damping is assumed for the plate). The damping enhancement is computed by (if ξ^{damp} is the damping ratio with active damping, and ξ the initial damping ratio):
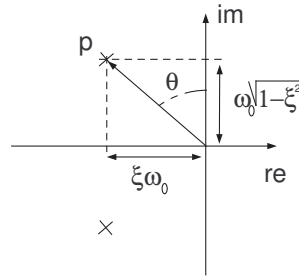
Figure 4.1: Gauss plane and representation of the poles
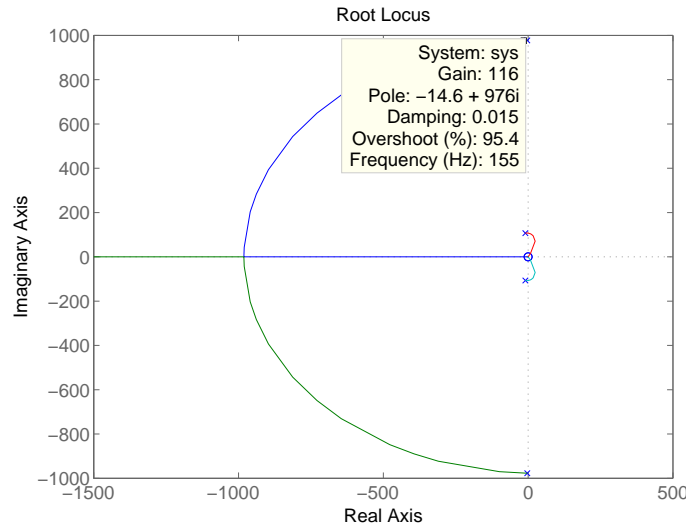
Figure 4.2: Root locus for the damping of one mode (output=$\dot{y}_1$)

$$20log_{10}(\frac{1}{2\xi}) - 20log_{10}(\frac{1}{2\xi^{damp}}) = 20log_{10}(\frac{\xi^{damp}}{\xi}) \qquad (4.2)$$

Here, the gain of damping is about $10dB$.

There is obviously a trade off between the efficiency of the active damping, and the safety of the loudspeaker. For commercial inertial actuators ([6]), the electromagnetic constant ϕ can reach 20 N/A (here it is $1,4$ N/A). So, with lower current we could have better efficiency. However, we are far from the destruction. More important feedback gain have been used in section 6, and performances of the active damping device are very good.

We mentioned earlier that velocity feedback on collocated system are unconditionally stable. However it is not the case in Fig.4.2, because the poles have a positive part if the gain is too important. To be entirely collocated, we must make a loop on the relative velocity $\dot{y}_1 - \dot{y}_2$ and not on $\dot{y}_1$ only. As detailed earlier, $\dot{y}_1$ is the velocity of the plate and $\dot{y}_2$ those of the loudspeaker. So Eq. 4.1 becomes

$$U = -g(\dot{y}_1 - \dot{y}_2) = -g(\psi\dot{r} - \dot{y}_2) \qquad (4.3)$$

and the damping matrix C of section 2.3 becomes

$$[C] = \begin{bmatrix} \psi^2 c + \frac{\phi^2\psi^2}{R} + \frac{\psi^2\phi g}{R} & -(\psi c + \frac{\psi\phi^2}{R}) - \frac{\psi\phi g}{R} \\ -\frac{\psi c + \frac{\psi\phi^2}{R}}{m} + \frac{\psi\phi g}{mR} & \frac{c + \frac{\phi^2}{R}}{m} + \frac{\phi g}{mR} \end{bmatrix} \qquad (4.4)$$
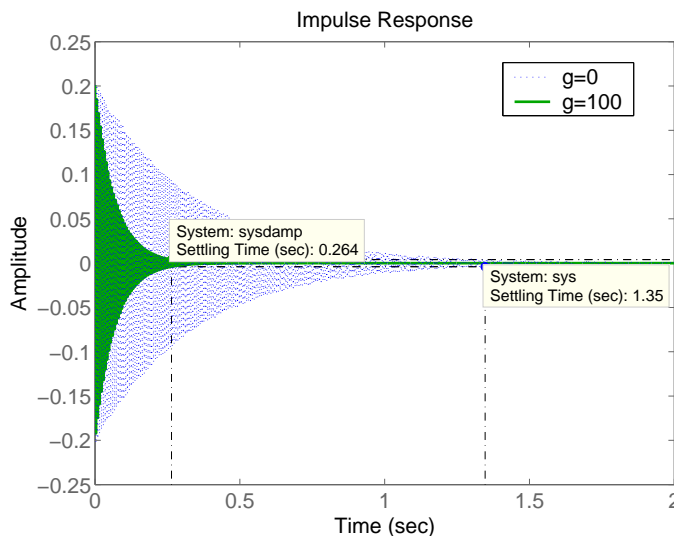
Figure 4.3: With and without state feedback
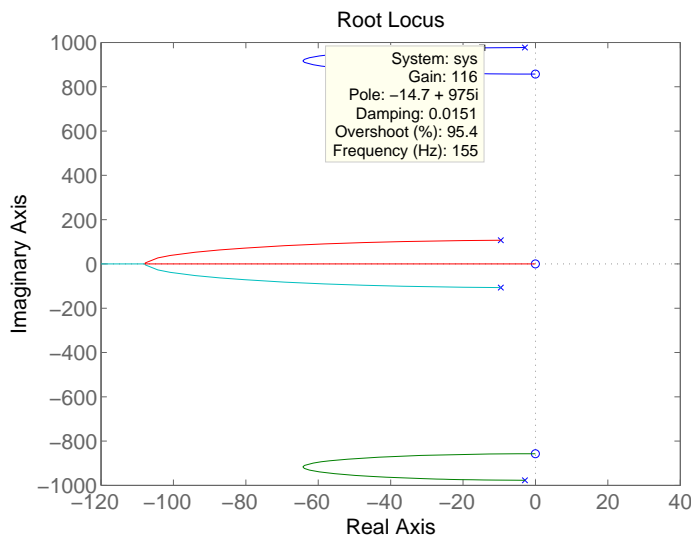
Figure 4.4: Root locus for the damping of one mode at $155Hz$ (output=$\dot{y}_1 - \dot{y}_2$)

Root locus is represented in Fig.4.4. In [11], this kind of root locus is typical for velocity state feedback. If we represent the same root locus for differents modes (here 59, 79, 115 and $155Hz$), we can see (Fig. 4.5) that for all the loop in the root locus, we have an infinite gain margin and so an unconditional stability. We can also emphasize the fact that, even if the gain is very important, the damping is not always better. There is a critical value of g which gives the maximum damping. In our application, we can not reach this critical value, because the voltage would be too important and so the loudspeaker could suffer...

To conclude this section, let us remember that theoretically, the collocation guarantees stability for any value of g, and that for all the modes inside a given bandwidth. However, the damping ratio is the same for low value of g, whatever the feedback ($\dot{y}_1$ or $\dot{y}_1 - \dot{y}_2$) . This is emphasized in Fig.4.2 and 4.4 : for a gain $g = 116$, damping is worth 1.5% for $\dot{y}_1$ feedback, and 1.51% for $\dot{y}_1 - \dot{y}_2$ feedback.
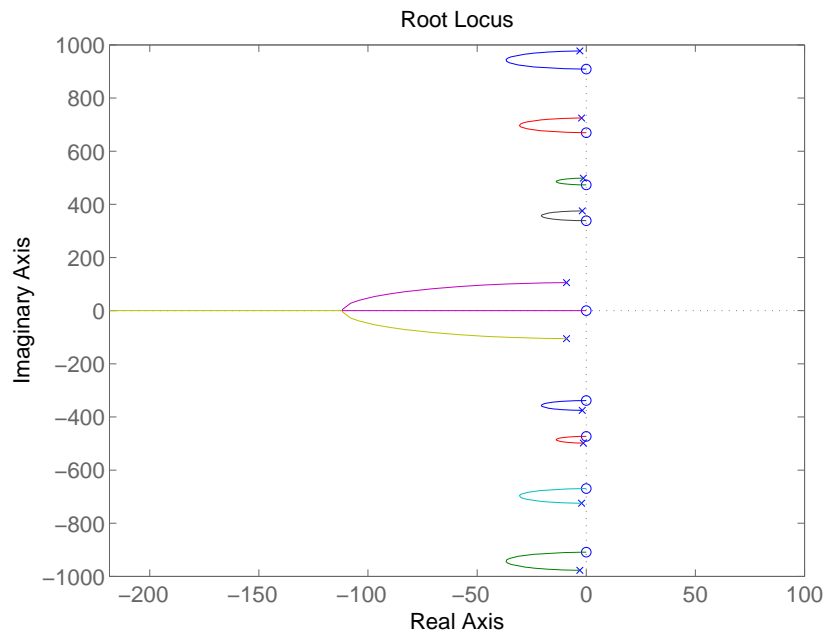
Figure 4.5: Root locus for the damping of several modes at 59, 79, 115 and $155Hz$ (output=$\dot{y}_1 - \dot{y}_2$)

4.2   Experimental validation

At the opposite of control made by state space feedback (section 6), the simple feedback allows theoretically a damping for all the modes. The location of the actuator has been discussed earlier at section 3.1.5.

4.2.1   Measurement system

Fig.4.6 and 4.7 presents the setup used during the experimental campaign. The plate is located on two small little pieces of foam. With a plate in an horizontal position, it is easier to connect the actuator. One `B&K` accelerometer is put on the plate, and the other on the magnet of the loudspeaker (both $B\&K\ 4393$). We use the associated charge amplifier `B&K 2635` because they allow to measure a velocity, and to use a high pass filter with a cut-off frequency of $10Hz$, avoiding the feedthrough component (the DC component) which could be fatal for the actuator! The sensibility is of $1V/(m/s)$ when it is used in velocity.

We choose a sampling frequency of $10000Hz$ for our application. This choice could seems too important, because the maximum controlled frequency is about $150Hz$: we will comment it in the section 6.3. To avoid the aliasing effect, an analogical lowpass filter `KEMO` is adjusted to $5000Hz$ (Shannon's theorem, see [15] or [16]).

Calculations will be made by means of a computer and an acquisition card `NI PCI-6035E`. It is commanded by a `c++` program, given in appendices (`simpleloop.cpp` in section A). The difference between the two sensors is made by the computer, then multiplied by a constant gain g and sent to the loudspeaker under voltage form. This command enters a `Velleman` power amplifier of constant gain, fixed to 22.

All the dynamics of the different parts of the loop are given in Appendix B.

Figure 4.6: Experimental setup (Photography)
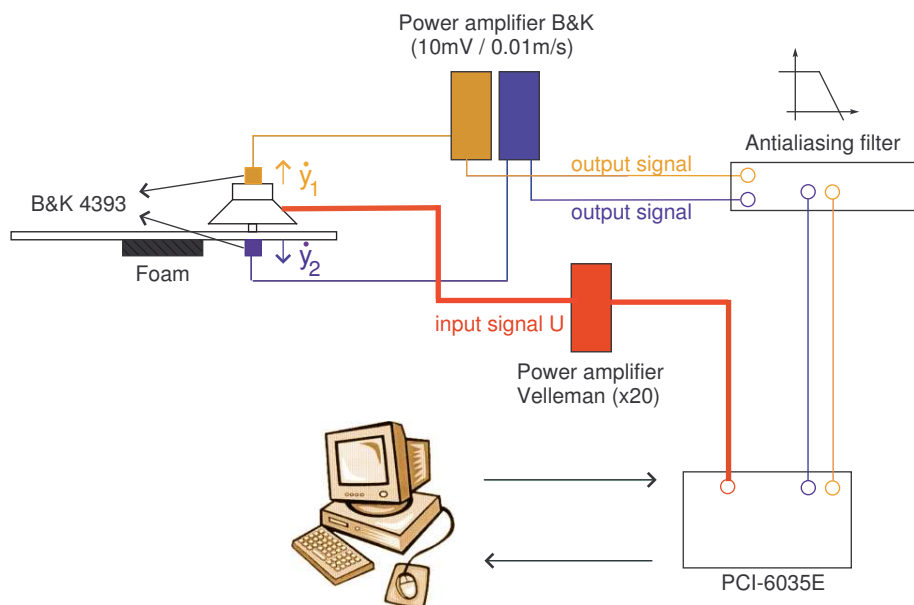
Figure 4.7: Experimental setup (Principle)

4.2.2   Results

Figure 4.8 shows us the efficiency of this simple method, for two different gain. All the modes are damped, more or less efficiently. If the loudspeaker is not located in a maximum amplitude of vibration, the effect is less efficient (like $165Hz$). At the opposite, $120Hz$ and $150Hz$ (which are 115 and $155Hz$ modes from the `Ansys` modeling) are well damped ($7dB$ and $11dB$

respectively, for $g = 110$).

Low frequencies are not examined in this work, because the behavior of the speaker is not linear, filter of pre-amplifier $B\&K$ has a cut-off frequency of $10Hz$, and modes of the plate are not well emphasized below $70Hz$. The last reason is the $50Hz$ noise, unavoidable in our application.

The amplification that we can see at Fig.4.9 is due to the phase lag inherent to the measuring chain, especially the KEMO filter. Section 6.3 explain in detail this problem. Because of this phase lag, the closed loop is not unconditionally stable, so the gain margin is not infinite anymore. A phase lead controller could be implemented into the closed loop, to enhance the closed loop bandwidth. However, we must keep in mind that there is always a trade-off between gain and phase margin in the design of such controllers (see [11]). Their design depends on the application : here we do not implent them.
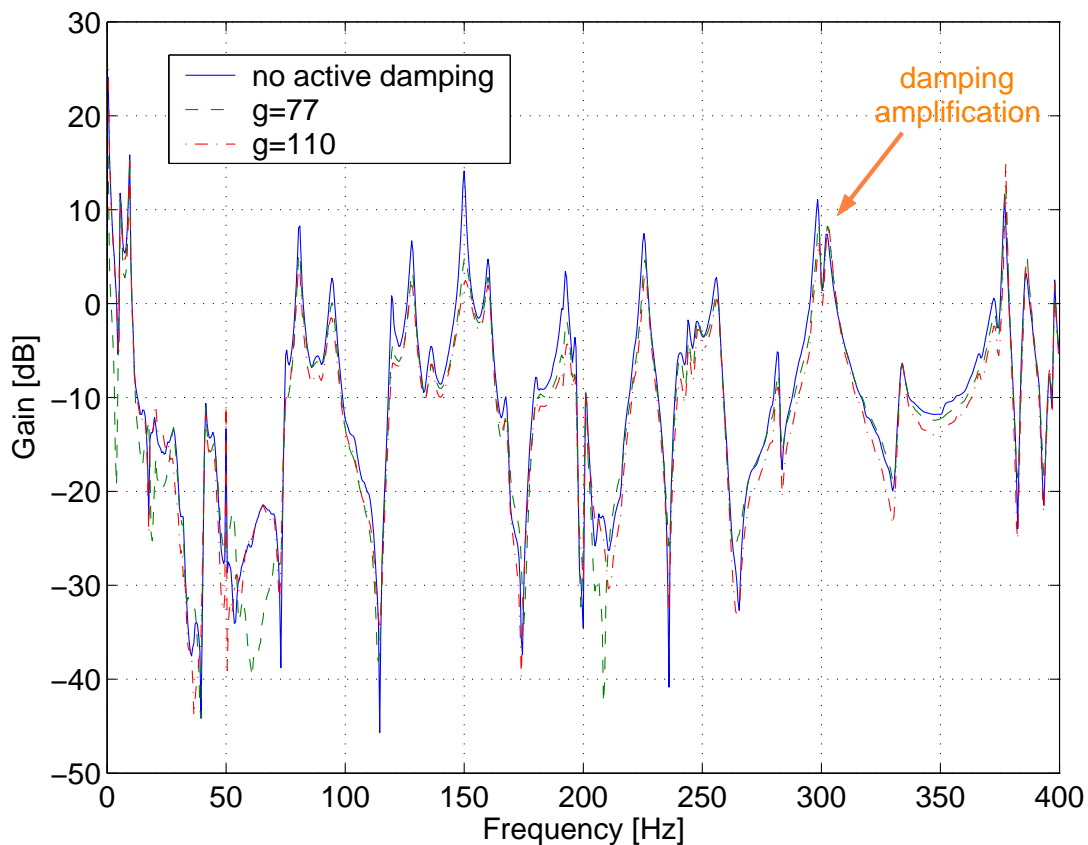
Figure 4.8: Effect of simple collocation on FRF (plate:55 - hammer:16)
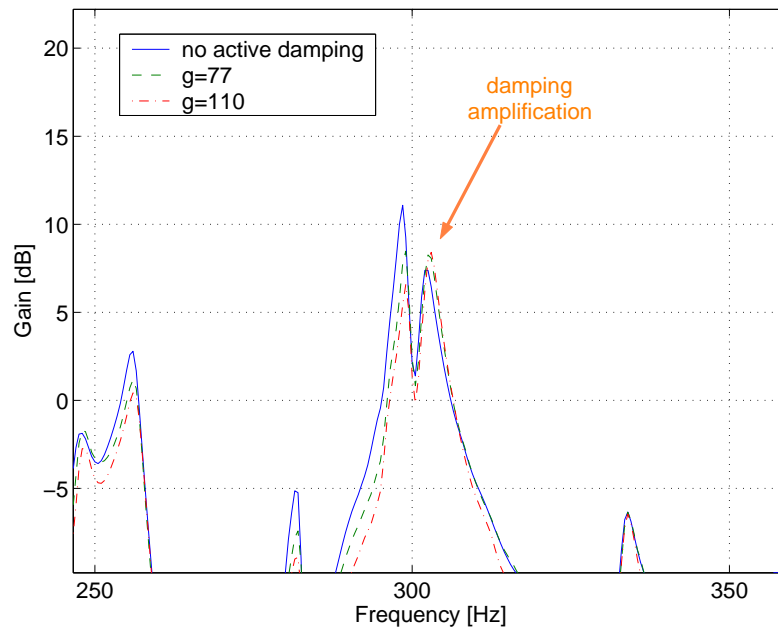
Figure 4.9: Zoom of Fig.4.8

4.2.3 Conclusion

To conclude, the simple feedback is a good approach in active damping system. All the modes are damped, no modeling of the loop is needed, and unconditionally stability can be reached thanks to collocation of the sensors and the actuator. However, a limitation in gain feedback exists because of the presence of unavoidable phase lags, due to the measuring chain. So a high sampling frequency is recommended, which could lead to some problems for high frequency applications.

Chapter 5

State space feedback approach

State space feedback approach is a convenient formulation for control in case of *MIMO* control loop. At each time, we find the value of a state vector, representing the physical characteristics of the system. This approach is very powerful : we will see later that, with pole placement method, we can fix the eigenvalue of the system where we want, and so its dynamics, under certain assumptions. This theory is developed in [17] and [18]. The first step of this method is to rewrite the whole system under a first order form : this is made in section 5.1.

5.1   State representation

The equation of motion of a linear mechanical system can be written

$$M\ddot{\vec{q}} + C\dot{\vec{q}} + K\vec{q} = \vec{F} \tag{5.1}$$

where M, C and K are the mass, damping and stiffness matrices of the system. $\vec{q}$ is a vector containing all the degrees of freedom of the system, and F is a column vector representing the applied forces. In state space feedback approach, these equations are rearranged as a first order system :

$$\dot{\vec{x}} = A\vec{x} + B\vec{u} \tag{5.2}$$
$$\vec{y} = C\vec{x} + D\vec{u} \tag{5.3}$$

Vector $\vec{x}$ represents now $[\vec{q}\ \dot{\vec{q}}]^T$, and is called the *state vector* (dimension $n \times 1$). In our case, it is composed of the displacement and the velocity of the considered output. u contains the command of the system (dimension $m \times 1$, where m is the number of actuators). $\vec{y}$ represents measurements of the system, and its dimension are $p \times 1$ if we have p measures for the plant. A is the system matrix ($n \times n$), B the command matrix ($n \times m$), C the output matrix ($p \times m$) and D the feedthrough matrix ($p \times m$) . Matrices A and B are depend on the system, while C and D are function of the measurement system, in other words of what can be measured from the plant. In most cases, D is 0 and C is composed of 0 and 1, or other constants. We have :

$$[A] = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \quad [B] = \begin{bmatrix} 0 \\ M^{-1}F \end{bmatrix}$$

A more visual expression of this system of equations is given in Fig.5.1. The control law is simply given by $u = -K\vec{x}$, where K is a line matrix in the case of a single-input multi-output system.

The dynamics of the system is fixed by the eigenvalue of the matrix $(sI - A)$ in Laplace's domain (see Ref. [18]). By using a state feedback of gain K, we modify these eigenvalues so
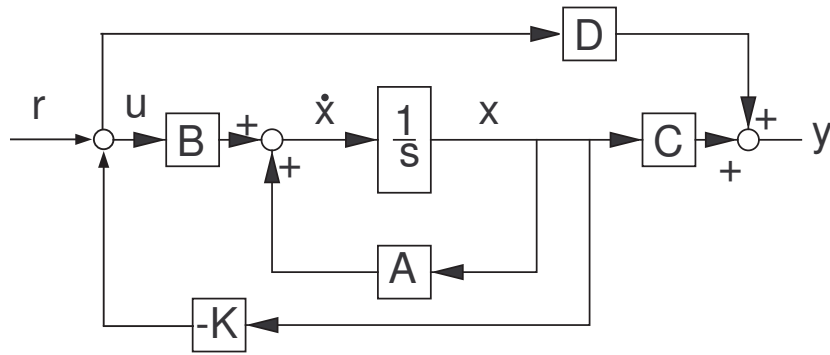
Figure 5.1: State feedback approach

that the state matrix of the system can be rewritten as $(sI - A + BK)$. So, by choosing a well-adapted matrix K, we can place the eigenvalue of the system where we want, to obtain good transient performances.

5.2 Control in state space approach

In a general way, two methods are used in *MIMO* control. The first on consists in a *pole placement*, i.e. that pole of the closed loop system can be located anywhere, especially with a chosen damping ratio. The second regulator is called *Linear Quadratic Regulator* is more complex, and takes into account a ponderation of the different states. They are briefly detailed in section 5.2.1 and 5.2.2. Let us note that the vectors are not written anymore, and consider that all the equations are vectorial ones.

5.2.1 Pole placement method

Placement pole method is used to modify algebraically the place of the pole in the Re-Im plane. If the system is controllable, then the poles p_i can be placed arbitrary. It is well-known that, in the Gauss plane (Fig.4.1), we have theses relations :

$$p \quad = \quad -\xi\omega_0 \pm j\omega_0\sqrt{1 - \xi^2} \tag{5.4}$$
$$\omega_0 \quad = \quad |p| \tag{5.5}$$
$$\xi \quad = \quad arcsin(\theta) \tag{5.6}$$

It means that, in our active damping application, the choice of the eigenvalue of the matrix $(sI - A + BK)$ allows us to fix the damping ratio of the plate, for the considered mode! With this reasoning, it would be easy to reach the ideal $\xi = 0.7$... However, and we emphasize that in section 4, the feedback gain can reach prohibited value! In this concrete case, we will have to choose the eigenvalues in function of the dynamic characteristics of the loudspeaker.

5.2.2 Linear Quadratic Regulator

A cost function is minimized, taking into account the output vector y (sensor measurement in section 5.1) and the command vector u. This function is the following

$$J = \int_0^\infty (y^T W_1 y + \rho u^T W_2 u) \, dt \tag{5.7}$$

Using relation 5.3 and assuming that matrix D is null, Eq. 5.7 can be rewritten in this way, using state variables x

$$J = \int_0^\infty (x^T C^T W_1 C x + \rho u^T W_2 u) \, dt \tag{5.8}$$

In a general way, $C^T W_1 C = Q$ and $\rho W_2 = R$. The Q matrix is semi positive definite, which implies that some of the states may be irrelevant. R is strictly positive definite, which means that any control has a cost (see Ref. [17]). The resolution of this problem is mathematically complex, and needs the solution of the algebraic Riccati equation. Details can be found in the previous reference, or in Ref. [19].

The practical interpretation of LQR method is that, the more Q matrix is important, the more the state variable are going to stay close of their equilibrium. But the bad side of the compromise is that the command must increase to satisfy this condition. In other words, the ratio between Q and R gives a priority either to the command cost ($q_{ii}/r_{ii} >>$ means an actuator more solicitated), or to the system reaction cost ($q_{ii}/r_{ii} <<$ means low actuator control, and large oscillations of the state variable around their equilibrium).

5.3   Observer problem

In a real plant, only some state variables can be actually measured. So, it is necessary to estimate the other ones. It is possible to reconstruct the state vector by means of a so called *state estimator* or *observer*. The idea is to build the state vector from input data (u) and output data (y). Before examining the equation of the observer, we can remark on Fig.5.2(a) its insertion in the control loop. The state feedback gain is then applied on the reconstructed vector ($u = -Kx^e$, with x^e the estimated states). Let us remark that an *observability* condition is required to make this state reconstruction. Details are given in section 5.4.

The equation of the estimate state are (5.9) :

$$\dot{x}^e = Ax^e + Bu; \qquad y^e = Cx^e \tag{5.9}$$

Because of process noise and modeling errors, the reliability of $\dot{x}^e$ should decrease unless on-line corrections are made, based on incoming measurement y. This is the idea developed by the equation 5.10 (called the *observer* equation):

$$\dot{x}^e = Ax^e + Bu + L(y - y^e) \tag{5.10}$$

L is an $n \times p$ matrix, weighting the difference between the real output of the plant and its estimation by the observer. When the estimated state is available, we can use it in the state feedback as in Fig.5.2(b). Then, if n is the dimension of the system, we must add n equations due to the state estimator.

In [17], one shows that for such a loop, the separation principle can be applied, and so the eigenvalue of the estimator have to be placed in a proper way. A rule of thumb is to design observer poles to be somehow faster than regulator ones (3 or 4 times faster, see [17]). Physically, it means that the estimated vector must be reconstructed faster than the dynamic of the system. Two methods allow to find values of matrix L. Once again we can use a pole placement method, or a *Kalman*'s filter developed in 5.5.2. The combination of a linear quadratic regulator and a *Kalman*'s filter is called Linear Quadratic Gaussian control method.

(a) Insertion of the state estimator

(b) Detail of the state estimator
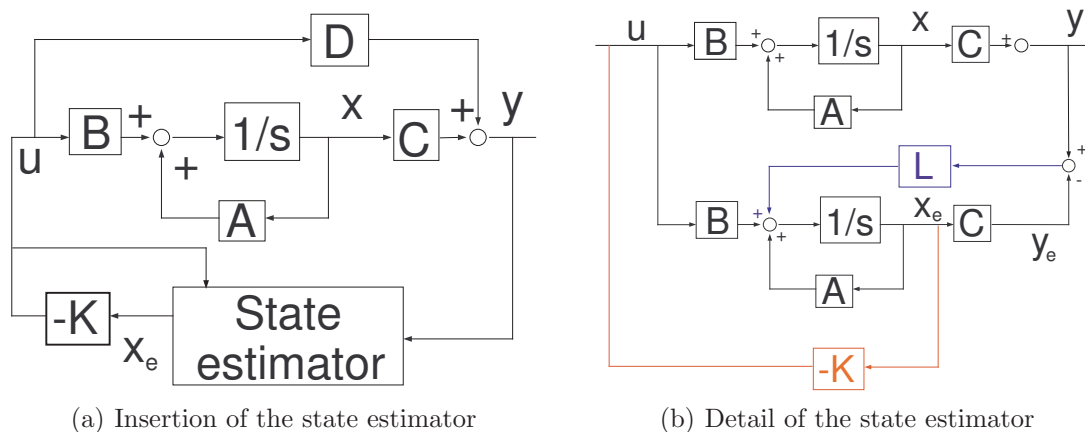
Figure 5.2: State estimator in the closed loop

5.4   Controllability and observability

A system is said controllable if it is possible to create a command u able to modify the system from an initial state x_i to a final state x_f in a finite time. In the same way, a system is said fully observable if the output of the plant y and the input u allow the determination of the state x, in a finite time. These two conditions (controllability 5.11 and observability 5.12) must be satisfied to develop the feedback of the system (n being the dimension of the state space representation).

$$rank\ \ [B\ \ \ AB\ \ \ A^2B\ \ \ ...\ \ \ A^{n-1}B] = n \tag{5.11}$$

$$rank\ \ [C^T\ \ \ A^T C^T\ \ \ A^{T^2} C^T\ \ \ ...\ \ \ A^{T^{(n-1)}} C^T] = n \tag{5.12}$$

5.5 Application to an inverted pendulum

This example is an introduction to the different concepts detailed in the previous section. The aim of the exercise is to create a regulator, i.e. to bring back the state variable to zero. The first step is the development of a regulator, without observer. The second step, more realistic, takes into account that all the state variables can not be observed, and illustrates the tuning of an observer.

5.5.1 Without observer

The two control methods have been implemented on the example of the inverted pendulum, composed by a rolling base and a rod which must stay in a vertical position (Fig.5.3). We must apply the correct force u allowing this vertical position. There is no damping assumed in the problem, and no modeling of the actuator.
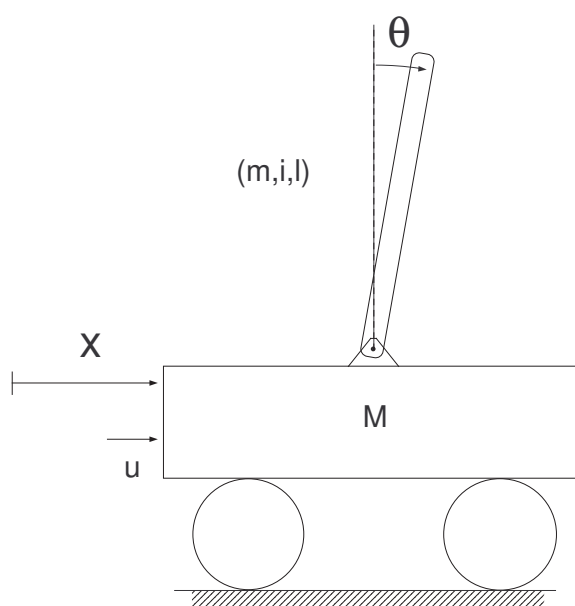
Figure 5.3: Model of the inverted pendulum

Here are the physical data of the problem :

- g=9.81 m/s

- m_{rod}=0.1 kg

- M=2 kg

- $I_{rod}=m \cdot i^2$=0.025 $kg.m$

- l_{rod}=0.5 m

In order to simplify the writing of the equations, we pose the following constant :

- $Mt=M + m_{rod}$=2.1 kg

- $L=\frac{I_{rod}+m_{rod}\cdot l_{rod}^2}{l_{rod}\cdot m_{rod}} = 1\ m$

- $\alpha = \frac{1}{1-\frac{m_{rod}\cdot l_{rod}}{Mt\cdot L}} = 1.0244$

This two degrees of freedom system is governed by two equations (5.13 and 5.14):

$$(1 - \frac{ml}{M_tL}cos(\theta)^2)\ddot{X} + \frac{ml}{M_tL}gsin(\theta)cos(\theta) - \frac{ml}{M_tL}sin(\theta)(\dot{\theta})^2 = \frac{1}{M_t}u \quad (5.13)$$

$$(1 - \frac{ml}{M_tL}cos(\theta)^2)\ddot{\theta} - \frac{g}{L}sin(\theta) + \frac{ml}{M_tL}sin(\theta)cos(\theta)(\dot{\theta})^2 = -\frac{1}{M_tL}cos(\theta)u \quad (5.14)$$

Theory of state space feedback is correct for linear systems; however, we see clearly with Eq. 5.13 and 5.14 that this problem is highly non-linear. So the first step is to linearize around the desired position $\theta = 0$. The previous equations become

$$(1 - \frac{ml}{M_tL})\ddot{X} - \frac{mlg}{M_tL}\theta = \frac{1}{M_t}u \quad (5.15)$$

$$(1 - \frac{ml}{M_tL})\ddot{\theta} - \frac{g}{L}\theta = -\frac{1}{M_tL}u \quad (5.16)$$

State variables are gathered in the vector x (pay attention to X which is the translational degree of freedom, and $\vec{x}$ which represents the state variables)

$$\vec{x} = \begin{pmatrix} X \\ \theta \\ \dot{X} \\ \dot{\theta} \end{pmatrix}$$

and matrices A, B, C and D are represented below :

$$[A] = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\alpha\frac{ml}{M_tL}g & 0 & 0 \\ 0 & \alpha\frac{g}{L} & 0 & 0 \end{bmatrix} \quad [B] = \begin{bmatrix} 0 \\ 0 \\ \frac{\alpha}{M_t} \\ -\frac{\alpha}{LM_t} \end{bmatrix} \quad [C] = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \quad [D] = [0]$$

The matrix C means that the output of the system corresponds to the state variable X. We compute the initial poles of the system :

```
poles =
  0
  0
  3.1701e+000
 -3.1701e+000
```

That shows clearly the instability of the initial system.

LQ regulator

The main parameters to fix are the Q and R matrices, which give an idea about the severity of the control. In order to illustrate this matter, we consider three matrices Q :

$$Q_1 = \left\{ \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix} \right\} \quad Q_2 = \left\{ \begin{matrix} 100 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 1 \end{matrix} \right\} \quad Q_3 = \left\{ \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 100 \end{matrix} \right\}$$

R is supposed to be a unit matrix (scalar in our case because there is one command). Q_1 corresponds to the same weighting of all the states, one privileges none of it. Q_2 corresponds

to a more important severity on the x and $\dot{x}$ states : in other words, we want a faster return to equilibrium position for these states. Q_3 is the same but for states θ and $\dot{\theta}$. Fig.5.4 represents the evolution of different states for an initial condition of $\theta = 1rad$, and for linearized equations of motion. In section 5.5.2, we implement this regulator (and its observer) for the non-linearized equations of motion.

With the `Matlab` function `LQR`, we can find the three state feedback gains (K_i is related to Q_i):

```
K1 = -1  -57.4   -2.9   -18.2
K2 = -10 -135.4  -18.3  -43.1
K3 = -1  -62.04  -3.07  -21.82
```

We see that with the Q_2 matrix (Fig.5.4(c)), the state x and its derivative must stay close to the equilibrium position, more than with Q_1 (Fig.5.4(a)) and Q_3 (Fig.5.4(e)). The consequence is that the force, acting in the x direction, becomes more important (Fig.5.4(d)).

On the other hand, there are not big differences in the behavior of the system in using Q_1 and Q_3 matrices. It seems obvious because there is no action on θ, so a weightening of this state is not determinant on the control process.

Now a word about the coefficient R. With LQ regulator, only the ratio q_{ii}/r_{ii} (diagonals elements of the Q and R matrices) is important.

- $q_{ii}/r_{ii} <<$ means a weak actuator, so the control is lower

- $q_{ii}/r_{ii} >>$ means a strong actuator, so the control is faster

This is represented in Fig.5.5(a), where $R = 0.1$ and Q_1 is used. Because the command can have higher value due to a weak weighting (Fig.5.5(b)), reaction of the system is faster than in Fig.5.4(a).

As expected an high weighting on the state variables forces them to stay close to the equilibrium position, and a weak weighting on the command allows large forces coming from the actuator.
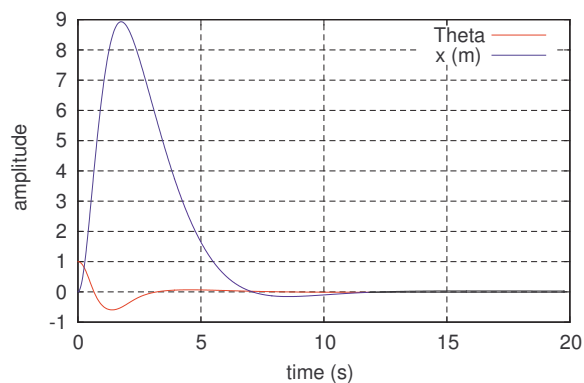
(a) Evolution with Q_1
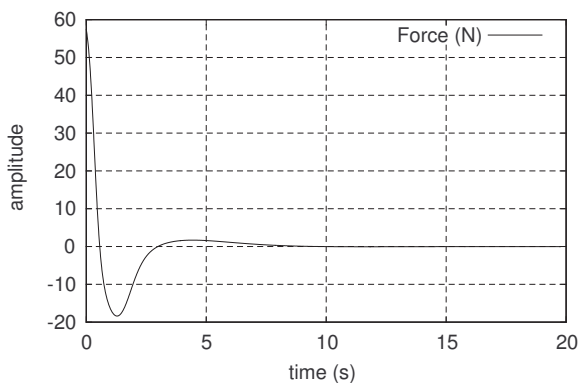
(b) Command associated to Q_1
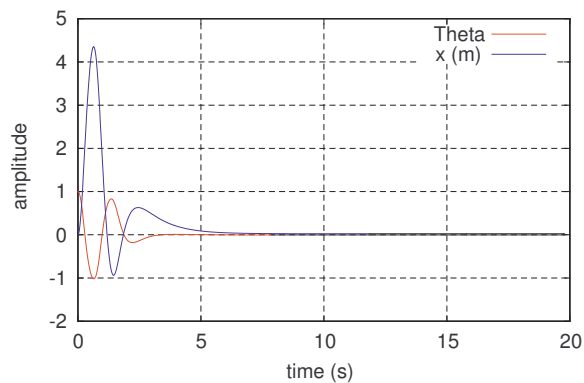
(c) Evolution with Q_2
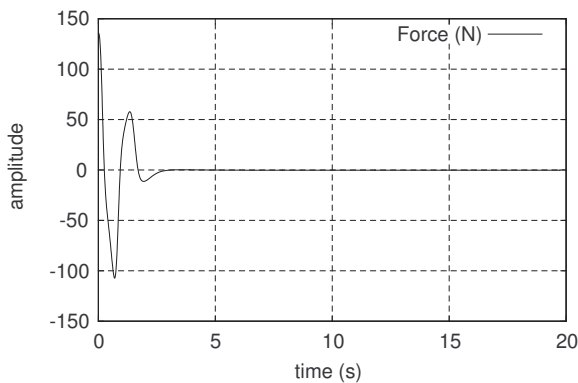
(d) Command associated to Q_2
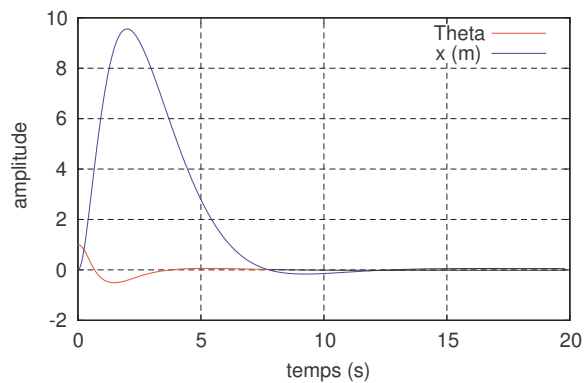
(e) Evolution with Q_3
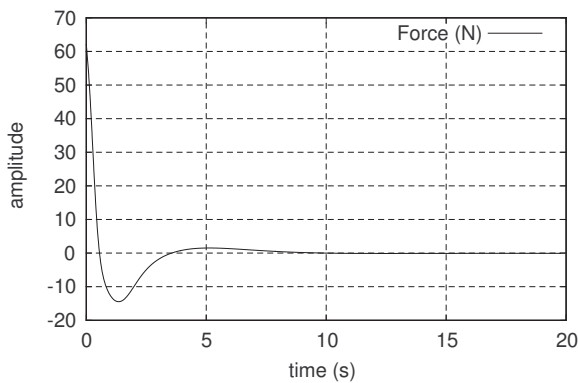
(f) Command associated to Q_3

Figure 5.4: Evolution of x, θ et u for different Q matrices (R=1)

(a) Evolution with Q_1 (b) Command associated to Q_1
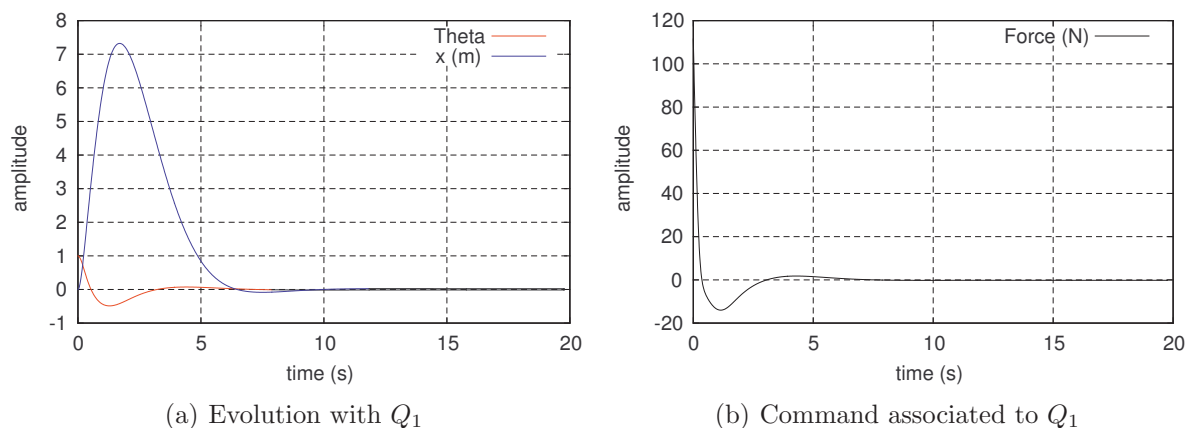
Figure 5.5: Evolution of x, θ and u with Q_1 and R=0.1

Pole placement control

As said earlier, we can impose the poles of the system, and thus its dynamics, if the controllability is verified. Reference [20] gives a solved example of the inverted pendulum, and the choice of the eigenvalue is the result of a compromise. Frequency of the poles influences the robustness of the system : if it is too weak, it is more robust but time to reach the set point is more important. At the opposite, a too high frequency increases dynamics of the system but gives some stability problems in practice, because noise effects are amplified. The last point is that a too weak damping conducts to an impossible convergence. We choose frequencies of 0.18 and $0.53Hz$ for the first simulation (superscript 1), and 0.1 and $0.4Hz$ for the second one (superscript 2). The damping ratio is equal to 0.9 in both cases. Our poles are thus :

$$p^1_{1,2} = -1.0179 \pm 0.4930i$$

$$p^1_{3,4} = -2.9971 \pm 1.4516i$$

$$p^2_{1,2} = -0.5655 \pm 0.2739i$$

$$p^2_{3,4} = -2.2619 \pm 1.0955i$$

The state feedback gains are computed thanks to the `place` command of `Matlab` :

```
KPP1 = -2.9642 -73.9361   -6.3198   -22.7811; %(0.18 and 0.53 Hz)
KPP2 = -0.5    -45.37      -1.87     -13.46;    %(0.1 and 0.4 Hz)
```

Without surprise, the first choice gives a faster reaction (Fig.5.6(a)) and consequently and higher command (Fig.5.6(b)).
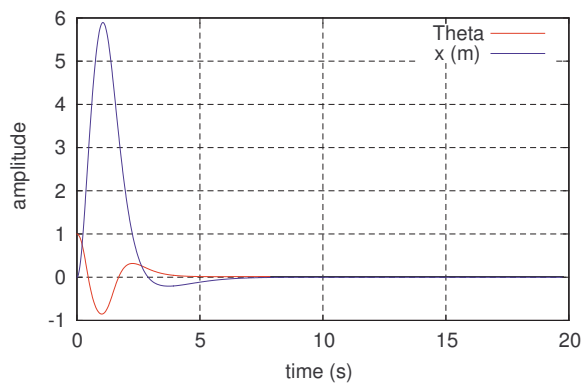
(a) 0.18 and 0.53 Hz
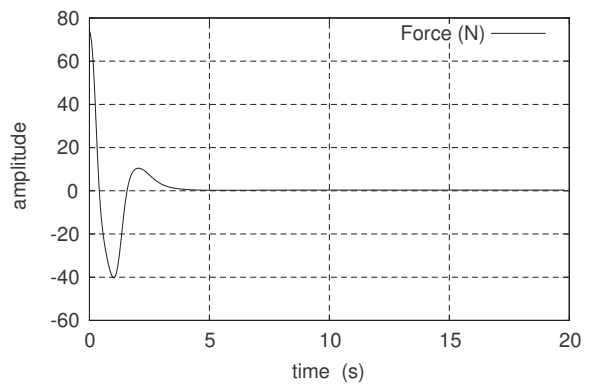
(b) Associated command
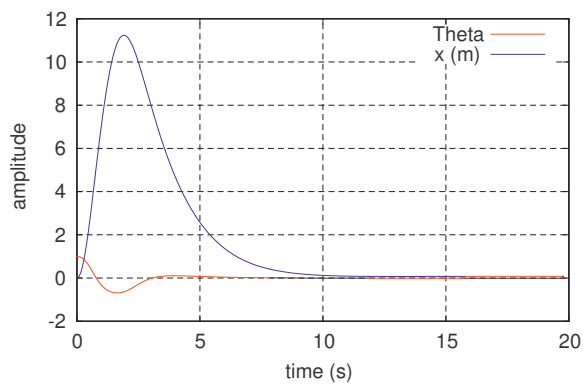
(c) 0.1 and 0.4 Hz
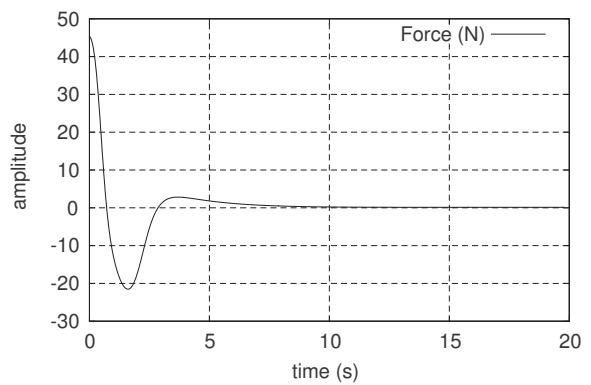
(d) Associated command

Figure 5.6: Evolution of x, θ and u for different poles

5.5.2   With observer

The development of the regulator is divided into two stages. First, the assumption that all the states are available is made and the research of the gain matrix K is made. For the same values of the system poles, we compute matrix K (K_{lqr} for the linear quadratic method and K_{pp} for pole placement) which gives :

```
KLQR =   -1.0000 -57.3616 -2.9145 -18.1816
KPP  =   -2.9642 -73.9361 -6.3198 -22.7811
```

The second step is to obtain the L matrix of the observer (see Eq. 5.10). *Kalman* filter allows a minimum sensitivity to disturbances. The considered equations of the system are

$$\dot{x} = Ax + Bu + Gw \tag{5.17}$$
$$y = Cx + Du + Hw + v \tag{5.18}$$

where w is the process noise, v the measurement noise. They are specified in terms of their variance matrices (defined by $E\{ww'\} = QN$, $E\{vv'\} = RN$, $E\{wv'\} = NN$,). Most of the time, $NN = 0$ because process and measurement noise are assumed to be uncorrelated. *Kalman* observer leads to an optimal observer, so that x^e it the best prediction of x in a statistical sense. In our example, $H = 0$, $G = [1\ 1\ 1\ 1]^T$ (noise on the command) and $QN = RN = 0.01$, which could represent a real value. We will not go further in the details of the *Kalman* method.

For the placement pole method, we chose the following values for the observer :

$$p_{1,2} = -4.5239 \pm 2.1910i$$

$$p_{3,4} = -5.0894 \pm 2.4649i$$

This poles corresponds to a frequency of 0.8 and $0.9Hz$, with a damping ratio $\xi = 0.9$. Even if the observer poles must be chosen with a frequency 3 to 4 times faster than 0.18 Hz and 0.53 Hz (design without observer), we can see in Fig.5.7(b) that the state are well estimated. This choice was made in a didactic goal, in order to remark the light delay of the observer before reaching the real states (Fig.5.7(b)).

The coefficients of the L matrix are :

```
LLQR =   7.3 26.8 360  1144  % Kalman's observer
LPP  =   19  159  3092 10071 % Observer designed by pole placement
```

The choice of the pole is maybe not possible in a real application : we notice the extremely high value of L matrices for the pole placement method. The equilibrium position is reached faster, but the command (not represented here) is disproportionate with respect to the LQR method.

The complete regulator (observer+regulator) was implemented and simulated for non-linearized equations of motion. The initial condition is now $\theta = 0.1rad$, $1rad$ being too far from the equilibrium position : consequently the regulator can not work properly, leading to a divergence in the simulation.

Fig.5.7(a) and 5.7(b) illustrates the response of X and θ state, as well as their estimation. We notice that faster the poles, faster the estimated variables reach the real state : thus better is the estimator. But pay attention that high coefficient of the L matrix can lead to amplification of the system noise. The system becomes less robust, so it can diverge more quickly.
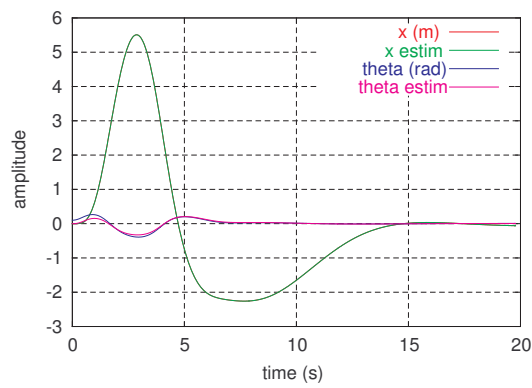
(a) LQR method
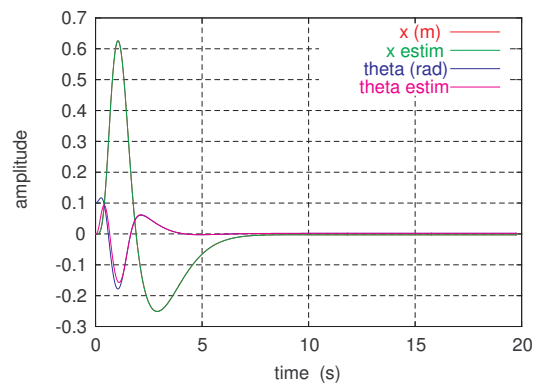
(b) Pole placement method

Figure 5.7: Evolution of the state variables and their estimation

Chapter 6

Application to the damped plate

6.1 Theory

6.1.1 Equations of motion

We assume that two modes are going to be controlled (one sensor and one actuator lead to a limited controllability and observability). The displacement of the plate can be expressed in modal coordinates, as $y_1 = \psi_1 r_1 + \psi_2 r_2$. Displacement of the loudspeaker is denoted by y. Equations of motion for two controlled modes gives (see details for one mode in section 2.3) :

$$
\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} \ddot{r}_1 \\ \ddot{r}_2 \\ \ddot{y} \end{pmatrix} + \begin{bmatrix} \psi_1^2(c + \frac{\phi^2}{R}) & \psi_1\psi_2(c + \frac{\phi^2}{R}) & -\psi_1(c + \frac{\phi^2}{R}) \\ \psi_1\psi_2(c + \frac{\phi^2}{R}) & \psi_2^2(c + \frac{\phi^2}{R}) & -\psi_2(c + \frac{\phi^2}{R}) \\ -\psi_1\frac{(c+\frac{\phi^2}{R})}{m} & -\psi_2\frac{(c+\frac{\phi^2}{R})}{m} & \frac{(c+\frac{\phi^2}{R})}{m} \end{bmatrix} \begin{pmatrix} \dot{r}_1 \\ \dot{r}_2 \\ \dot{y} \end{pmatrix}
$$

$$
+ \begin{bmatrix} \omega_1^2 + \psi_1^2 k & \psi_1\psi_2 k & -\psi_1 k \\ \psi_1\psi_2 k & \omega_2^2 + \psi_2^2 k & -\psi_2 k \\ -\frac{\psi_1 k}{m} & -\frac{\psi_2 k}{m} & \frac{k}{m} \end{bmatrix} \begin{pmatrix} r_1 \\ r_2 \\ y \end{pmatrix} = \begin{bmatrix} \psi_1\frac{\phi}{R} \\ \psi_2\frac{\phi}{R} \\ -\frac{\phi}{mR} \end{bmatrix} U \tag{6.1}
$$

with ψ_i related to the deformed shape at considered node, for the mode i; m, c and k the mass, the damping and the stiffness of the loudspeaker respectively. r_i is the modal weighting (in terms of displacement), and y is the displacement of the loudspeaker. Because of the unity mass normalization of `Ansys`, we have directly a diagonal stiffness matrix, gathering the square of natural pulsations.

For the first order arrangement, A and B matrices are created like in section 5.1, while $D = 0$ and C is representing the difference of velocities $\dot{y}_1 - \dot{y}$:

$$
C = \begin{bmatrix} 0 & 0 & 0 & \psi_1 & \psi_2 & -1 \end{bmatrix} \tag{6.2}
$$

6.1.2 Controllability and observability

We will restrict our study to two modes for controllability and observability reasons. A system is controllable if

$$
rank \ [B \ \ AB \ \ A^2B \ \ ... \ \ A^{n-1}B] = n \tag{6.3}
$$

and observable if

$$
rank \ [C^T \ \ A^T C^T \ \ A^{T^2} C^T \ \ ... \ \ A^{T(n-1)} C^T] = n \tag{6.4}
$$

where n is the dimension of the first order system : in our case, $n = 6$.

With a modeling of the loudspeaker and with two modes of the plate, the system observes these two conditions. However, if we had only one velocity sensor, the observability would be not verified for two modes, and the estimation of the state vector would be impossible. The same reasoning can be made for the controllability : it is obvious that a higher number of actuators leads to a more controllable system.

6.1.3   Poles of the undamped system

The poles of the system allows us to verify our equations :

```
sysCT=ss(A,B,C,D,'statename', states,'inputname', inputs,'outputname', outputs);
f=abs(pole(sysCT)/2/pi)
xi=-real(pole(sysCT))./abs(pole(sysCT))
```

which gives the original frequencies of the considered modes, as well as the damping ratio associated. Let us remember that there is no damping on the plate : those obtained here are caused by the loudspeaker.

```
f =
  1.5550e+002 % modes of the plate
  1.1536e+002
  1.5936e+001 % naturak frequency of the loudspeaker

xi =
  2.9944e-003 % dampind due to the glued loudspeaker
  2.8386e-003
  8.3013e-002 % damping ration of the loudspeaker
```

6.1.4   Sampling frequency

Before creating observer and regulator, we must convert the state-space model into a discrete form, using `c2d` command (under `Matlab`), and specifying the sampling frequency of the A/D converter (NI 6035E). For reasons that we detail in section 6.3, we choose a sampling frequency of $10kHz$.

6.1.5   Equation in discrete time

One can demonstrate that, in DT, the matrices A and B are depending on the sampling period T_e. System of equation 5.2 and 5.3, like those of the observer 5.10 are written in tho following form :

$$x_{k+1} = Gx_k + Hu_k \tag{6.5}$$
$$y_k = Cx_k + Du_k \tag{6.6}$$
$$x_{k+1}^e = Gx^e + Hu_k + L(y_k - y_k^e) \tag{6.7}$$
$$y_k^e = Cx_k^e + Du_k \tag{6.8}$$

where x^e refers to the estimated state, and x_k to the sample of the state vector, at time $t = kT_e$. C and D matrices are unchanged, while A and B are recomputed according to the sampling period. Details are given in [18].

6.1.6   Observer and regulator design

Observer

As mentioned earlier (section 5.3), an observer is needed to have an estimation of the state vector. Two observers have been designed :

1. The first one, by pole placement : in the literature ([21],[17],...) it is advised to have poles at least three times faster than the poles of the system. We chose poles of the observer at frequencies of 466, 482 and $497Hz$. The damping ratios are the same as before. It can appear weak, but after some simulations, we remark that this value induces the minimum settling time of the closed-loop system; in other words the system reach its static equilibrium faster.

2. The second observer is developed as a *Kalman*'s filter : parameters were detailed in section 5.5.2 : we take $G = B$ in Eq. 5.17 and $H = 0$ in Eq. 5.18. As for the linear quadratic regulator, the important parameter for the filter is the ratio between diagonal element of matrices QN and RN : if it is important, it means that the system is very noisy, and so the filter allows a few informations : consequently it is "lower". At the opposite, a high ratio gives a faster filter, but the noise is amplified. We choose $QN = 100$ and $RN = 1$ because noise is very present in our application.

In the experimental part, we remark that pole placement method were not efficient, because the noise of the system is very present. *Kalman*'s filter gives best estimations of the states, so we consider this observer in the following.

The `dlqe` command allows us to find the L matrix of the observer, called here L_{kalm}.

```
Lkalm =
  -1.7111e-10
   2.0311e-10
   5.8445e-10
   1.9290e-01
  -2.2895e-01
  -6.5894e-01
```

Poles of the observer are the following (poles P_{Kalm} given in discrete time):

```
Pkalm =
   0.0093
   0.9957 + 0.0895i
   0.9957 - 0.0895i
   0.9976 + 0.0641i
   0.9976 - 0.0641i
   0.9999
```

The design of the observer allows a few information from the process. The filter is "low" : frequency of the above poles are 0.14, 102, 142 and $7445Hz$. We see here that their frequencies is not $3 - 4$ times faster : at the opposite, the first three poles are under those of the system!

Experimentally, we will see that noise is so present that a quick observer gives us back states which are flooded into the noise, so the observation seems to be a white noise : the control is thus impossible. The *Kalman*'s filter, lower, can estimate the two modes. In section 6.2.2, we

will check this *Kalman*'s observer.

Theoretically, both of the observers are working properly, with a light advantage to those designed by pole placement, because it is faster. And a better estimation allows a better control of the closed loop. However, we will show only the results of the simulation for the closed loop with a *Kalman*'s obeserver.

Regulator

We fix the damping ratio of the two modes at 2%, and at 50% for the loudspeaker. Using the `place` command of MATLAB, we find the feedback gain matrix K_{regDT}. (Poles are created in continuous time, and transformed in numerical poles by the relation $z_i = e^{p_i T_e}$)

```
KregDT =
    5.3628e+03
   -2.0516e+04
    1.2252e+02
    1.2217e+02
   -1.3436e+02
   -1.3001e+02
```

The last step is to close the system, including the observer and the regulator. The following command lines are used :

```
regDT=reg(sysDT,KregDT,Lkalm);
%reg() to insert the regulator and the observer in the plant
clDT=feedback(sysDT,regDT,1);
%feedback() to close the loop, 1 for negative feedback
```

All the poles of the system are, of course, inside the unit circle, as represented in Fig.6.1. The damping increases when the poles are close to the origin of the plane, and their frequency is higher if the argument of the pole is important. A pole located in $(1, 0)$ means that we have an integrator in the system. Here we do not have any integrators.

We can have a simulation of an unit impulse on the input of the plant, and analyze the behavior of both output ($\dot{y}_1$ concerning the plate and $\dot{y}$ the loudspeaker). Fig.6.2 shows us the initial system (blue) and the damped one (green). We see that the damping works properly : the settling time is $0.152s$ for the closed loop, while it is $1.14s$ for the open loop. Corresponding Bode curve is represented at 6.3, where we can see the damping efficiency ($16dB$ for the design of this section, and $20dB$ for a damping ratio of 3% for the modes). The file `control.m` in Appendices A gives the command used to design the regulator and the observer.
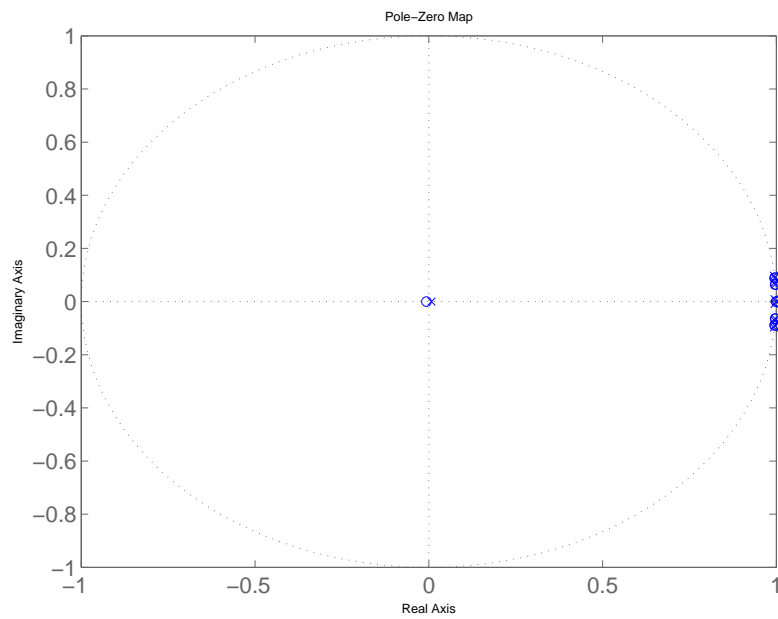
Figure 6.1: Poles and zeros of the closed loop system
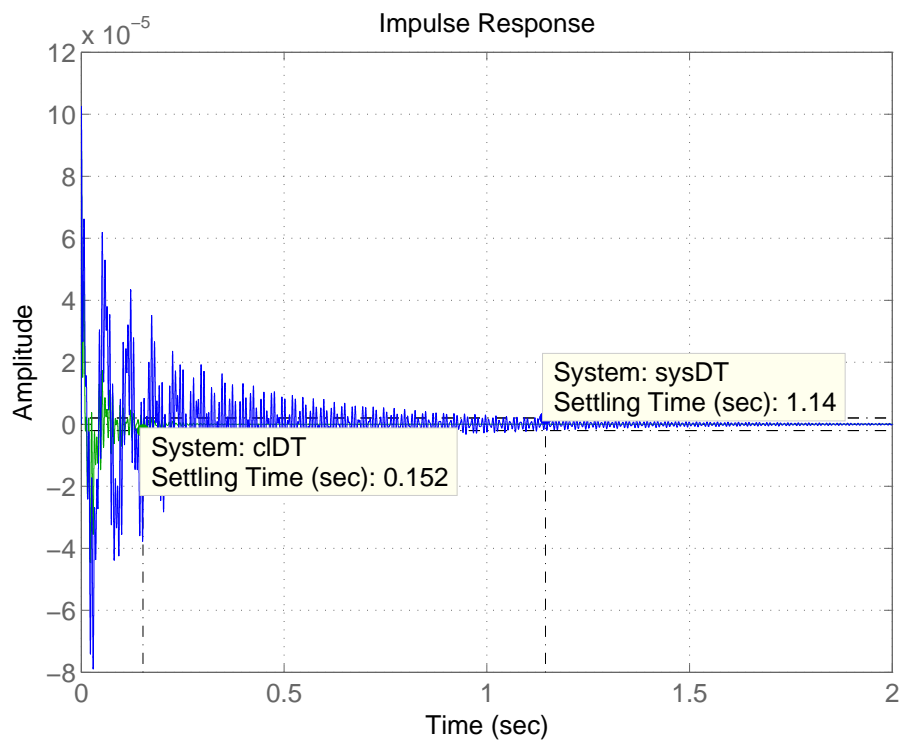
Figure 6.2: Closed loop reaction of the plant to a unit impulse on the input (damped system in green)
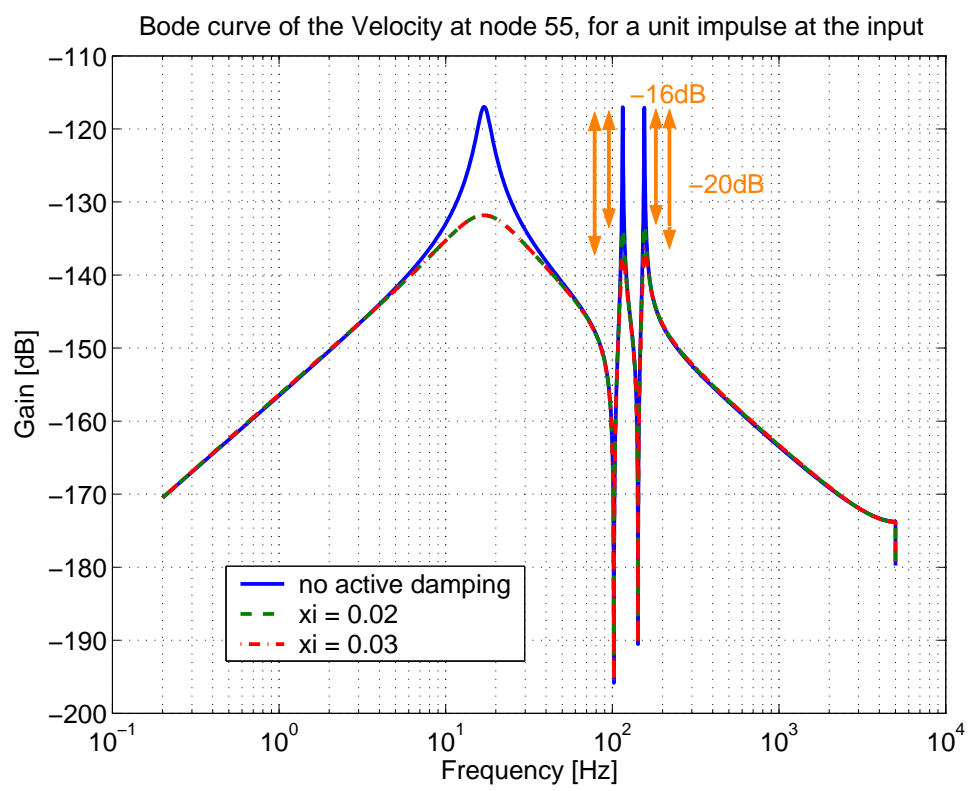
Figure 6.3: Bode curve of Fig.6.2

6.2   Experimental campaign

6.2.1   Parameters

Setup is the same as in Fig.4.7. Equations implemented in the computer are the following, and are deduced from 6.7 and 6.8 :

$$x_{k+1}^e = (G - LC)x_k^e + Hu_k + Ly_k \tag{6.9}$$

The Table 6.1 summarize parameters used in our experiments. The first experiment will be made with an observer designed by *Kalman*'s filter and a regulator by pole placement, with $\xi_{modes} = 0.02$. The second one will be the same, with $\xi_{modes} = 0.03$.

	Observer parameters	Regulator parameters
First experiment	$QN = 100$, $RN = 1$ $G = B$	pole placement $\xi_{modes} = 0.02$ $\xi_{speaker} = 0.5$
Second experiment	$QN = 100$, $RN = 1$ $G = B$	linear quadratic $\xi_{modes} = 0.03$ $\xi_{speaker} = 0.5$

Table 6.1: Parameters for active damping

With the choice that we made in Table 6.1, we suppose that the system is very noisy, so a few informations can pass through the filter.

6.2.2   Checking the observer

The observer is supposed being able to reconstitute an unknown state. Our estimated state vector is given by

$$x^e = \begin{pmatrix} r_1^e \\ r_2^e \\ y_{speaker}^e \\ \dot{r}_1^e \\ \dot{r}_2^e \\ \dot{y}_{speaker}^e \end{pmatrix} \tag{6.10}$$

where r_i^e is the modal weighting, and y is the displacement of the loudspeaker.

In order to check the observer, an impact hammer is made on the plate. Sensibility of $B\&K$ accelerometers is $1V/(m/s)$, with a pre-filtering at $10Hz$, and cut off frequency of $10kHz$. Fig.6.4 shows us that states 4 and 5 of x^e (i.e. the modes at 120 and $150Hz$, in terms of velocity) are observed, and that *Kalman*'s observer acts like a pass-band filter.

The second mode (fifth state variable in Eq.6.2.2) is very well observed. Nevertheless, several parasitic modes are detected for the $120Hz$ mode (the fourth state in Eq.6.2.2). These modes are very close of the desired one, and so it is impossible to filter them. It seems clear that an enhancement of the state estimator would lead to a better efficiency of the active damping. However, the effect is beneficial, because others modes are going to be damped, while the observability and controllability matrix allow only two modes in the design.

The choice of a *Kalman*'s filter is not innocent : an estimator has been designed with pole placement method in the previous section. As mentioned in the literature, we take the poles of
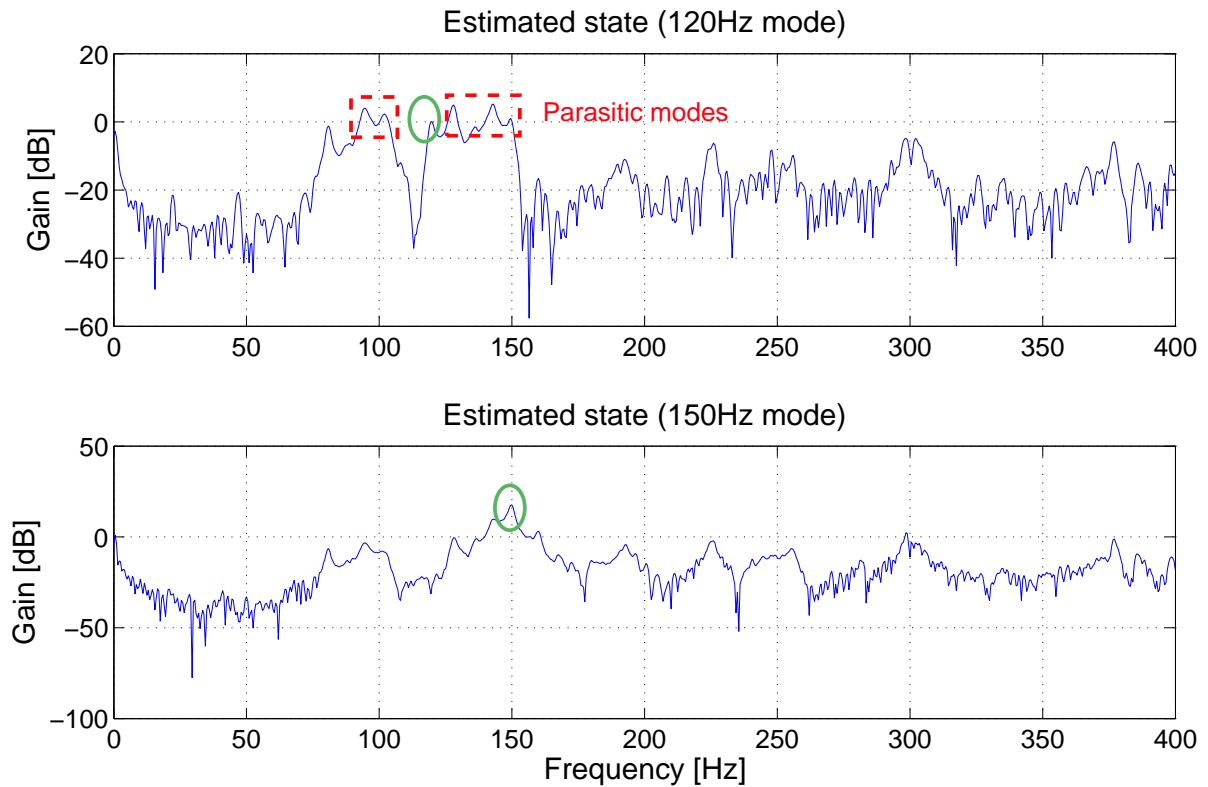
Figure 6.4: Observed state with *Kalman*'s designed observer (estimated modes circled in green)

this observer three times higher than the higher pole of the system. Damping ratios were chosen similar to those of the system. Unfortunately, with such an observer, the peaks of Fig.6.4 were drowned into the noise.

6.2.3   Closing the loop

Conditions of the experiment are the following :

- sensibility of accelerometers (integration made by the pre-amplifier) : $1V/(m/s)$;

- pre-filtering : $10Hz$;

- cut-off frequency of the amplifier $B\&K$: $10kHz$;

- cut-off frequency of the Anti-aliasing filter : $5kHz$;

- hammer impact on node 16, and measured velocity at node 55;

- we perform a mean on four frequency response for each curve presented in this section.

As the power amplifier used is not adjustable, it is wiser to set a saturation at the output of the acquisition card, and so avoid that an instability leads to the destruction of the actuator. We suppose a maximum at $0.2V$, so the output of the amplifier can not go over $4.5V$.

In Fig.6.5 and Fig.6.6, we use the regulator designed by pole placement. Fig.6.5 gives an overview of the general level of damping in the frequency band $0 - 400Hz$. Ideally, if the

observer was correctly designed, only theses modes would be damped.

Fig. 6.6 is a zoom of the considered modes. The damping seems very efficient for the considered modes. A verification of our modeling ca be made by comparing the attenuation obtained in the simulation and experimentally (see Eq. 4.2). Table 6.2 gives the comparison. As far as the experimental part is concerned, estimation of ξ without active damping is made thanks to a peak picking method.

	Simulation		Experiment	
	$120Hz$	$150Hz$	$120Hz$	$150Hz$
$\xi_{modes} = 0.02$	16	16	11	15
$\xi_{modes} = 0.03$	20	20	13	23

Table 6.2: Attenuation (in dB) with state feedback

The simulation seems sometimes far from the reality. Many errors of modeling are in cause : foam while ANSYS gives the results for free-free conditions, the plate is slightly folded, inaccuracy in the dynamics modeling of the loudspeaker, no damping assumed in theoretical approach.
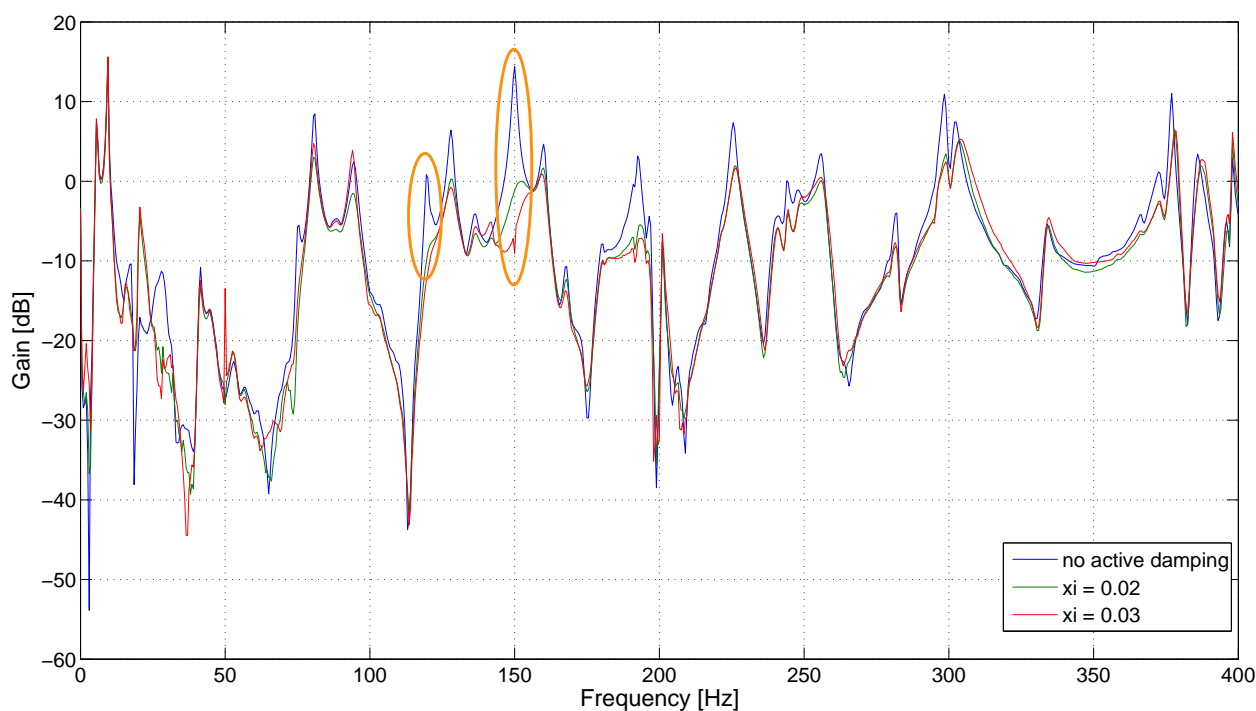
Figure 6.5: Active damping (regulator designed by pole placement - observer by *Kalman*'s filter)
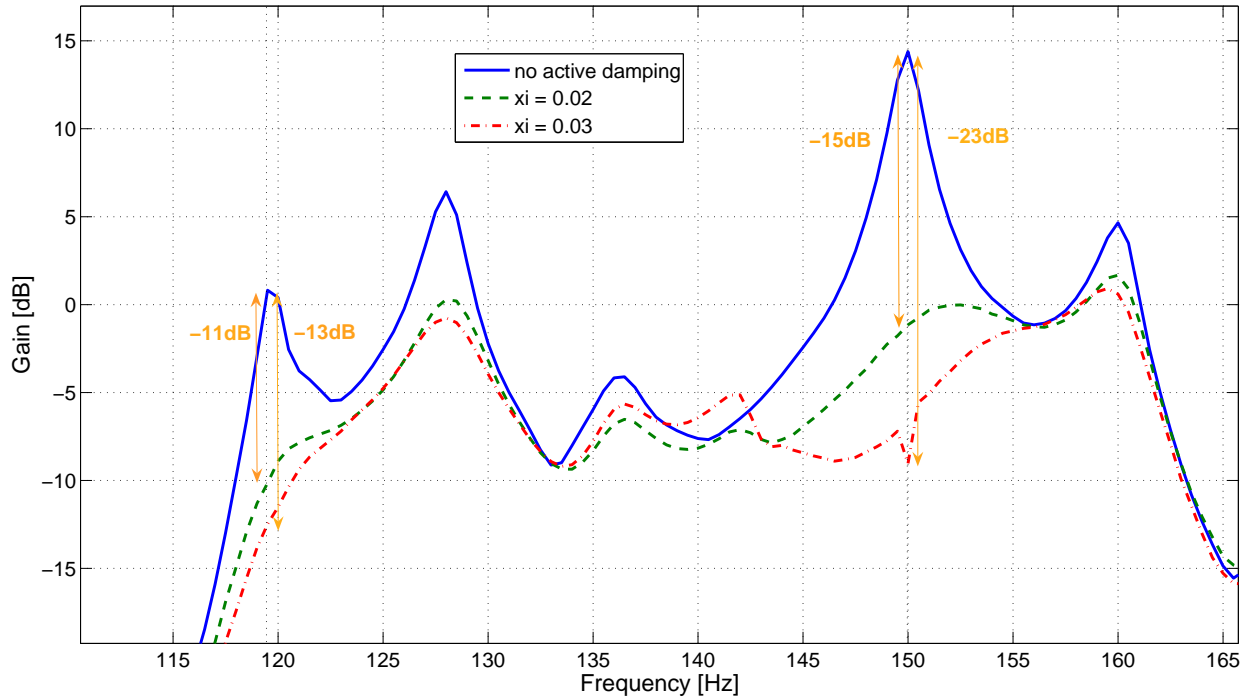
Figure 6.6: Zoom on concerned modes

6.3   Delays in the loop

The last chapter of this work will be consecrated to the delays in the control loop. In a general way, an analogical filter is required to avoid aliasing effect, and to allow a correct A/D conversion. The KEMO filter used is very efficient in terms of roll-off (order 8!) but it has a cost in terms of phase lag. As illustrated in Appendix B (Fig. 2), the phase is decreasing very quickly. At the beginning of the experimental campaign, we choose a sampling frequency of $2000Hz$, and a cut-off frequency of $200Hz$ for the KEMO filter. It means that for the $120Hz$ mode, we were not far of a $180°$ phase lag, which leads to an unstable system. With a sampling frequency of $10kHz$, we have a phase lag of about $5-6°$.

Fig. 6.7 illustrates this phenomena. The control is made by a simple return, as in section 4. The gain is fixed to $g = 77$ for the two acquisitions. We can see that mode at $190Hz$ is amplified if the anti-aliasing filter is set to $2kHz$, while the active damping stay efficient up to $377Hz$ (not represented). However, the phase lag is only about $10-11°$ at this frequency, so the system would be stable up to an higher frequency, maybe 1000 or $2000Hz$. In fact, the pre-amplifier $B\&K$ also has a phase lag, due to its pass band filter ($10Hz$-$10kHz$), so as the power amplifier and the acquisition card.

Appendix B shows the bode curve (amplitude only) of the pre-amplifier. We have a slope of $+40dB/dec$ before the cut-off frequency of $10Hz$, and a roll-off of $-20dB/dec$ after it. So, we can imagine the corresponding phase lag happening between the bandwidth $100-1000Hz$, and conclude that this phase lag is certainly not negligible.
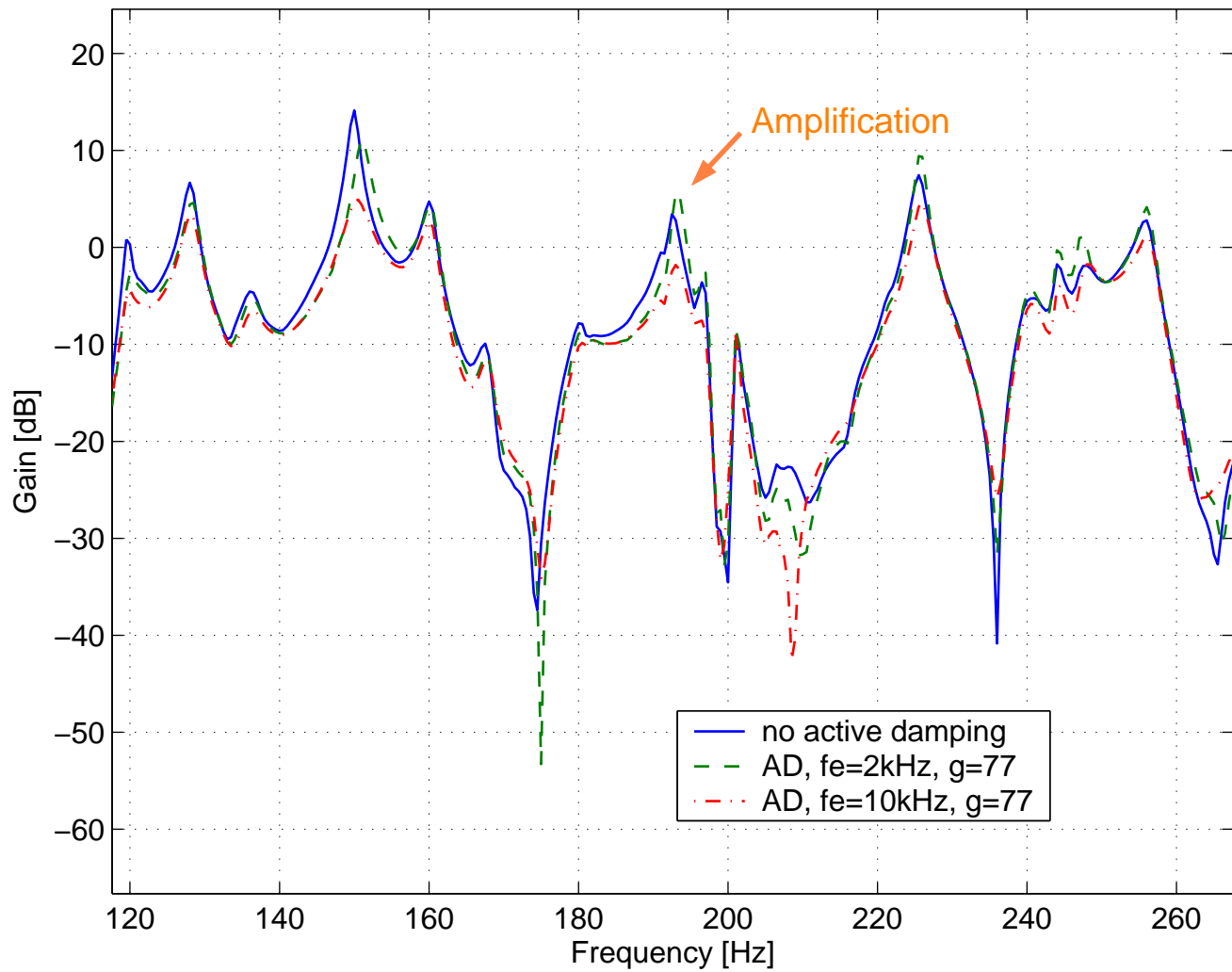
Figure 6.7: Amplification of the vibration due to phase lag in the loop

Chapter 7

Conclusion

An active damping device composed by a plate and a loudspeaker has been developed in this work. Starting with totally unknown components, the first step was to identify the natural frequencies of the plate, as well as the deformed shapes. As far as the loudspeaker is concerned, we have identified its electromechanical characteristics, assimilated it to a two degrees of freedom system composed by a spring, a damper and an active electrical element.

The second step was to describe the equations of motion for the coupled system, and to theoretically study the effect of a sensor-actuator collocation thanks to a root locus. We verified the unconditional stability for all the considered modes of the plate. This kind of control is very simple, because no model of the structure is necessary. All the modes are damped, as far as the actuator is not located at a node of vibration.

As third step, we developed a more complex method of control, based on the state space feedback approach. This method can be very powerful as the pole placement method allows us to set the pole of the closed loop system anywhere (of course taking into account the limits of the actuator). The main drawback is due to controllability and observability : to control a lot of modes, an important number of sensors and actuators is needed. This method is also less easy to implement than the previous one, because we need a complete model of the system. Interest of this method for our active damping application can seem limited but it allowed us to have a better understanding of different concepts such as the Kalman's filter, the state space representation or the LQG method.

Finally, we have implemented these two methods on the actual system. Both have met the same limitations : not an infinite gain margin, while it must be the case for collocation. This is mainly due to the different phase lags of elements in the experimental chain, principally because of the anti-aliasing filter (order 8 means good roll-off, but important phase lag). Having control of the acquisition card requires time. Implementing the control loop demands stringency : forgetting a gain or bad-typing an element of the matrix can lead to prohibited voltage on the speaker and destroy it.

The results obtained between experimental and numerical approaches agree well, with an attenuation of $20dB$ for the $150Hz$ mode. But the control loop could be improved. The first enhancement concerns the modeling of the plate : our active damping does not involve of low and high frequency modes. They can be modeled by adding a term in the transfer function between the plate and the speaker. The second enhancement affects a modeling of the experimental chain, especially the anti-aliasing filter. It would allow us to identify the real controllable bandwidth. Moreover, we could have an idea of the real phase and gain margins. Finally, the last enhancement is the development of an accurate observer : experience has

shown that the mode at $120Hz$ was not properly reconstructed.

The purpose of this very interesting work was to illustrate a basic introduction on active damping. Many aspects have been explored, as mechanics, electronics, control systems and data processing. All these aspects are deeply close to mechatronics, and emphasize the variety of applications proposed to modern engineers.

Bibliography

[1] J. B. Hunt. *Dynamic vibration absobers*. Mechanical Engineering Publications LTD London, 1979.

[2] D. A. Bies and T. M. Yang. Hybrid vibration isolation system for helicopters. *US Nav. Res. Lab. Shaock and Vibr. Bull.*, 37:39–47, 1968.

[3] Wikipedia - the free encyclopedia. http://en.wikipedia.org.

[4] W. Bolton. *Mechatronics*. Longman group, 1995.

[5] G. C. Onwubolu. *Mechatronics - principles and application*. Elsevier, 2005.

[6] Micromega Dynamics sa. Micromega dynamics. http://www.micromega-dynamics.com.

[7] CSA Engineering Inc. Csa engineering. http://www.csaengineering.com.

[8] C. Paulitsch, P. Gardonio, S.J. Elliot, and R. Boonen. Design of a ligthweight, electro-dynamic, inertial actuator with integrated velocity sensor for active vibration control of a thin lightly- damped panel. In *Acvitve vibration control and smart structures*. Proceedings of ISMA, 2004.

[9] D. K. Lindner, G. A. Zvonar, and D. Borojevic. Performance and control of proof-mass actuators accounting for stroke saturation. *AIAA Journal of Guidance, Dynamics and Control*, 17(5):1103–1008, September - October 1994.

[10] D. K. Lindner, G. A. Zvonar, and D. Borojevic. Nonlinear control of a proof-mass actuator. *AIAA Journal of Guidance, Dynamics and Control*, 20(3):464–470, May-June 1997.

[11] A. Preumont. *Vibration Control of Active Structures - An Introduction*. Kluwer Academic, 1997.

[12] M. McEver and D. J. Leo. Autonomous vibration suppression using on-line pole-zero identification. *Journal of Vibrations and Acoustics*, 123:487–495, October 2001.

[13] J. Q. Sun. Some observations on physical duality and colocation of structural control sensor and actuators. *Journal of Sound and Vibration*, 194(5):765–770, 1996.

[14] C. Conti and S. Boucher. Vibrations mcanique. Facult Polytechnique de Mons, 2004.

[15] P. Dehombreux. Mesure et analyse de signaux vibratoires. Facult Polytechnique de Mons, 2000.

[16] R. Longchamp. *Commande numrique des systmes dynamiques*. Presses Polytechniques et Universitaires Romandes, 1995.

[17] P. Albertos and A. Sala. *Multivariable Control Systems*. Springer, 2004.

[18] C. Renotte. Commande des systmes multivariables. Facult Polytechnique de Mons, 2005.

[19] E. Weisstein. Wolfram mathworld. http://mathworld.wolfram.com/RiccatiDifferentialEquation.html.

[20] M. Dahleh, M. A. Dahleh, and G. Verghese. Lectures on dynamics systems and control. Massachussets Institute of Technology.

[21] O. Verlinden and G. Kouroussis. Easydyn example : Roller on a tilting track, simulation and control. Technical report, Facult Polytechnique de Mons, 2003.

Appendix A

Listings

File simpleloop.cpp

C++ file executing the simple loop feedback. Sampling frequency is $10kHz$

```
1    #include "nidaqex.h"
     #include "fstream.h"
     #include "iostream.h"


     /*
      * Main:
      */

10   void main(void)
     {
         /*
          * Local Variable Declarations:
          */

         i16 iStatus = 0;
         i16 iRetVal = 0;
         i16 iDevice = 1;
         i16 iChan = 0;
20       i16 iGain = 1;
         f64 dSampRate = 100000.0;
         f64 dScanRate = 20000.0; //(two times the sampling) because double buffer
         u32 ulCount = 8; // length of the buffer
         f64 dGainAdjust = 1.0;
         f64 dOffset = 0.0;
         i16 iUnits = 0;
         i16 iSampTB = 0;
         u16 uSampInt = 0;
         i16 iScanTB = 0;
30       u16 uScanInt = 0;
         static i16 piBuffer[2000] = {0};
         static f64 Y[2000] = {0.0};
         static i16 piHalfBuffer[1000] = {0};
         i16 iHalfReady = 0;
         i16 iDAQstopped = 0;
         u32 ulPtsTfr = 0;
         u32 ulRetrieved = 0;
         i16 iNumMUXBrds = 0;
         i16 iNumChans = 2;
40       static i16 piChanVect[2] = {0, 1};
         static i16 piGainVect[2] = {1, 1};
         i16 iIgnoreWarning = 0;
         i32 lTimeout = 180;
         i16 iYieldON = 1;
         f64 output = 0.0;
         i16 i=0;

         iStatus = Timeout_Config(iDevice, lTimeout);
         iRetVal = NIDAQErrorHandler(iStatus, "Timeout_Config",iIgnoreWarning);
50
             // Config Double Buffer

             DAQ_Config (iDevice, 0, 0); /* internal trigger & timing */
             iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Config", iIgnoreWarning);
```

```
        DAQ_DB_Config (iDevice, TRUE);   /* enable double buffering */
        iRetVal = NIDAQErrorHandler(iStatus, "DAQ_DB_Config", iIgnoreWarning);

        // Config Rates

        iStatus = DAQ_Rate(dSampRate, iUnits, &iSampTB, &uSampInt);
        iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Rate",iIgnoreWarning);

        iStatus = DAQ_Rate(dScanRate, iUnits, &iScanTB, &uScanInt);
        iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Rate",iIgnoreWarning);

        // Config Channel

        iStatus = SCAN_Setup(iDevice, iNumChans, piChanVect, piGainVect);

        iRetVal = NIDAQErrorHandler(iStatus, "SCAN_Setup", iIgnoreWarning);

        // Start Acquisition

        iStatus = SCAN_Start(iDevice, piBuffer, ulCount, iSampTB,
        uSampInt, iScanTB, uScanInt);

        iRetVal = NIDAQErrorHandler(iStatus, "SCAN_Start",
        iIgnoreWarning);

        while (TRUE) {

        iStatus = DAQ_DB_HalfReady(iDevice, &iHalfReady,&iDAQstopped);

        if ((iHalfReady == 1))
                {

                iStatus = DAQ_DB_Transfer(iDevice, piHalfBuffer,
                &ulPtsTfr, &iDAQstopped);

                iStatus = DAQ_VScale(iDevice, iChan, iGain, dGainAdjust,
                dOffset, 2, piHalfBuffer, Y);
                        // Y[0] for plate velocity
                        // Y[1] for speaker velocity

                Y[0]=-Y[0]; // the accelerometers is glued upside down
                output=-10*(Y[0]-Y[1]); // here we adjust the feedback gain

                // Saturation : don't forget the Velleman x22!! (0.2x22=4.4V)
                // important because if the system is cabled in the wrong way...
                // loudspeaker could be destroyed

                f64 sat=0.2;

                        if (output>sat)
                        {
                                output=sat;
                        }

                        else if (output<-sat)
                        {
                                output=-sat;
                        }

                        iStatus = AO_VWrite(iDevice, 0,output);
```

```
                    // Writing the output value, in Volts, on the board

120           }

      }

  }

  /* End of program */
```

File statefeedback.cpp

C++ file for the state space feedback. Sampling frequency is $10kHz$

```
1    #include "nidaqex.h"
     #include "fstream.h"
     #include "iostream.h"


     /*
      * Main:
      */

10   void main(void)
     {
         /*
          * Local Variable Declarations:
          */

         i16 iStatus = 0;
         i16 iRetVal = 0;
         i16 iDevice = 1;
         i16 iChan = 0;
20       i16 iGain = 1;
         f64 dSampRate = 100000.0;
         f64 dScanRate = 20000.0;
         u32 ulCount = 12;
         // size of the complete buffer : we have 3 data : Y[0],Y[1] and Y[2]. Buffer is composed by
         // 4 packs of theses 3 data, classed in this way :
         // Y[0]|Y[1]|Y[2]|Y[0]|Y[1]|Y[2]||Y[0]|Y[1]|Y[2]|Y[0]|Y[1]|Y[2]|
         // (two half buffer)
         // we have a scanning frequency of 20000, and we take not all the packs, but one on two. So
         // the sampling rate is divided by two ==> 20000/2=10000Hz!
30       f64 dGainAdjust = 1.0;
         f64 dOffset = 0.0;
         i16 iUnits = 0;
         i16 iSampTB = 0;
         u16 uSampInt = 0;
         i16 iScanTB = 0;
         u16 uScanInt = 0;
         static i16 piBuffer[2000] = {0};
         static f64 Y[2000] = {0.0};
         static i16 piHalfBuffer[1000] = {0};
40       i16 iHalfReady = 0;
         i16 iDAQstopped = 0;
         u32 ulPtsTfr = 0;
         u32 ulRetrieved = 0;
         i16 iNumMUXBrds = 0;
         i16 iNumChans = 3;
         static i16 piChanVect[3] = {0, 1, 2};
         static i16 piGainVect[3] = {1, 1, 1};
         i16 iIgnoreWarning = 0;
         i32 lTimeout = 180;
50       i16 iYieldON = 1;
         f64 output = 0.0;
         f64 input = 0.0;


             iStatus = Timeout_Config(iDevice, lTimeout);
```

```
            iRetVal = NIDAQErrorHandler(iStatus, "Timeout_Config",iIgnoreWarning);

            // Config Double Buffer

60          DAQ_Config (iDevice, 0, 0);                          /* internal trigger & timing */
            iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Config", iIgnoreWarning);

            DAQ_DB_Config (iDevice, TRUE);           /* enable double buffering */
            iRetVal = NIDAQErrorHandler(iStatus, "DAQ_DB_Config", iIgnoreWarning);

            // Config Rates

            iStatus = DAQ_Rate(dSampRate, iUnits, &iSampTB, &uSampInt);
            iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Rate",iIgnoreWarning);
70
            iStatus = DAQ_Rate(dScanRate, iUnits, &iScanTB, &uScanInt);
            iRetVal = NIDAQErrorHandler(iStatus, "DAQ_Rate",iIgnoreWarning);

            // Config Channel

            iStatus = SCAN_Setup(iDevice, iNumChans, piChanVect, piGainVect);

            iRetVal = NIDAQErrorHandler(iStatus, "SCAN_Setup", iIgnoreWarning);

80          // Observability matrix C of the model

            f64 C[6]={0, 0, 0, 7.1400e-001 ,-8.4746e-001, -1};

            // Matrices for Fe=10000Hz
            // -----------------------

            // State matrix A of the model
            f64 G[6][6]={{ 9.9737e-001,  1.3492e-005,  1.6827e-005,  9.9892e-005,  2.4299e-008,  2.8696e-008},
                    { 1.3833e-005,  9.9523e-001, -1.9965e-005,  2.4299e-008,  9.9813e-005, -3.4047e-008},
90                  { 3.9844e-005, -4.6111e-005,  9.9994e-001,  6.9989e-008, -8.3042e-008,  9.9902e-005},
                    {-5.2479e+001,  2.6181e-001,  3.3622e-001,  9.9697e-001,  4.8990e-004,  5.7899e-004},
                    { 2.7202e-001, -9.5293e+001, -3.9878e-001,  4.8990e-004,  9.9467e-001, -6.8673e-004},
                    { 7.8415e-001, -8.9517e-001, -1.1495e+000,  1.4122e-003, -1.6750e-003,  9.9802e-001}};

            // Command matrix B of the model
            f64 H[6]={ 9.9818e-010,
                    -1.1843e-009,
                    -3.4113e-009,
                     1.9945e-005,
100                 -2.3656e-005,
                    -6.8192e-005};

            // Observer matrix
            // Q=100, R=1, Gnoise=B
            f64 L[6]={-1.7095e-010,
                     2.0306e-010,
                     5.8526e-010,
                     1.9290e-001,
                    -2.2895e-001,
110                 -6.5894e-001};
            // xi_modes=0.02, xi_speaker=0.5, gains /22 (because of power amplifier!)
            /*f64 K[6]={2.4376e+002,
                    -9.3256e+002,
                     5.5689e+000,
                     5.5534e+000,
                    -6.1074e+000,
                    -5.9098e+000};*/
```

```
          // xi_modes=0.03, xi_speaker=0.5, gains /22 (because of power amplifier!)
120       f64 K[6]={  1.2849e+002,
                     -1.7412e+003,
                      6.8614e+000,
                      8.8628e+000,
                     -9.6984e+000,
                     -5.9340e+000};

          f64 GLC[6][6]={0.0};  // Matrix G-LC computed in one times and not at each time step
          f64 Xest[6][2]={0.0}; // Estimate states : column 0 for time k, and column 1 for time k+1

130
          ofstream file("result.dat"); // Just a verification of the good computation of G-LC
          for (i16 i=0;i<6;i++)
          {
                  for (i16 j=0;j<6;j++)
                  {
                          GLC[i][j]=G[i][j]-L[i]*C[j];
                          file << " " << GLC[i][j] << "\n";
                  }
140       }
          file.close();

          // Start Acquisition

          iStatus = SCAN_Start(iDevice, piBuffer, ulCount, iSampTB,
          uSampInt, iScanTB, uScanInt);

          iRetVal = NIDAQErrorHandler(iStatus, "SCAN_Start",
          iIgnoreWarning);
150
          while (TRUE) {

          iStatus = DAQ_DB_HalfReady(iDevice, &iHalfReady, &iDAQstopped);


          if ((iHalfReady == 1))  // && (iStatus == 0)
                  {
160               iStatus = DAQ_DB_Transfer(iDevice, piHalfBuffer, &ulPtsTfr, &iDAQstopped);

                  iStatus = DAQ_VScale(iDevice, iChan, iGain, dGainAdjust,
                  dOffset, 3, piHalfBuffer, Y);
                          // Y[0] for the plate
                          // Y[1] for the speaker
                          // Y[2] for the command U

                          Xest[0][0]=Xest[0][1];
                          Xest[1][0]=Xest[1][1];
170                       Xest[2][0]=Xest[2][1];  // instant k+1 becomes instant k
                          Xest[3][0]=Xest[3][1];
                          Xest[4][0]=Xest[4][1];
                          Xest[5][0]=Xest[5][1];
                          Xest[0][1]=0.0;
                          Xest[1][1]=0.0;
                          Xest[2][1]=0.0;
                          Xest[3][1]=0.0;              // initialization of time k+1 for a new computation
                          Xest[4][1]=0.0;
                          Xest[5][1]=0.0;
180                       output = 0.0;              // output is set to zero at the beginning of the cycle
```

```
                            Y[0]=-Y[0]; // because the accelerometer is upside down on the plate

                            input=(Y[0]-Y[1]); // sensibility = 1V/m/s

                            for(i16 i=0;i<6;i++)
                            {
190                                 for(i16 j=0;j<6;j++)
                                    {
                                            Xest[i][1]+=GLC[i][j]*Xest[j][0];
                                    }
                                    Xest[i][1]+=H[i]*Y[2]+L[i]*input;
                                    //Xest[i][1]+=L[i]*input;
                                    // this line to verify the observer, without command

                                    output+=-(K[i]*Xest[i][1]); // state space feedback

200                         }

                            // Saturation

                            if (output>sat)
                            {
                                    output=sat;
                            }

                            else if (output<-sat)
210                         {
                                    output=-sat;
                            }


                            iStatus = AO_VWrite(iDevice, 0,output);
                            // writing on the PCI board the value of the command in voltage
                    }

            }
220


        }

        /* End of program */
```

File control.m

Design of the controller and observer

```
1   clear all
    close all
    clc

    load K % complete matrices for modes between 0 and 200Hz
    load M
    load modal
    load psi % deformed shape for all the nodes, at considered modes
    load freq % natural frequencies
10  freq'

    mode=[];
    dim=input('How many modes ?');
    for ii=1:1:dim
        num(ii)=input('Which mode ?');
    end


    n=0;
20  while (n<1||n>70)
    n=input('On which node do you want to place the speaker ? (1-70)');
    %n2=input('Second measure on the plate at node ?');
    end

    speaker=[];
    mode=[];



30  for index=1:length(num)
        speaker=[speaker psi(n,num(index))];
        mode=[mode freq(num(index))];
         for jndex=1:1:length(num)
            temp1(index,jndex)=M(num(index),num(jndex));
            temp2(index,jndex)=K(num(index),num(jndex));
         end
    end

    freq=mode;
40  M=temp1;
    K=temp2;

    w=freq*2*pi;

    % ------------------------- %
    %   Speaker characterization  %
    % ------------------------- %


50  f0=16;
    m=0.410;
    R=5;
    L=90e-6;
    c=7.5;
    k=4720;
```

```
    w=2*pi*f0;
    phi=1.4;
    xi=0.08;

60
    % ---------------------------- %
    %   Equations of the system      %
    % ---------------------------- %

    a=speaker; % contains modal amplitudes at considered node and considered mode
    d=c+phi^2/R;

    MM=[[M zeros(dim,1)];zeros(1,dim) 1];

70  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%        Building CC, KK         %%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    for ii=1:1:length(freq)
        for jj=1:1:length(freq)
            CC(ii,jj)=a(ii)*a(jj)*d;
            KKpetit(ii,jj)=a(ii)*a(jj)*k;

        end
    end
80
    for ii=1:1:length(freq)
        CCcol(ii)=-a(ii)*d;
        CClin(ii)=-a(ii)*d/m;
        KKcol(ii)=-a(ii)*k;
        KKlin(ii)=-a(ii)*k/m;
    end

    CC1=[CC CCcol'];
    CC2=[CClin d/m];
90  CC=[CC1;CC2]

    KK1=[temp2+KKpetit KKcol'];
    KK2=[KKlin k/m];
    KK=[KK1;KK2];

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%         Building FF              %%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

100 for ii=1:1:length(freq)
        FF(ii)=a(ii)*phi/R;
    end

    FF=[FF -phi/m/R]';

    %[lambda,vect] = poleMCK (MM,CC,KK);  % Verification of the system


    A=[zeros(dim+1) eye(dim+1);-inv(MM)*KK -inv(MM)*CC];
110 B=[zeros(dim+1,1);inv(MM)*FF];

    %C=[zeros(1,dim+1)  a 0;zeros(1,length(A)-1) 1];
    %D=[zeros(1,1);zeros(1,1)];

    C=[zeros(1,dim+1)  a -1];
    D=0;
```

```matlab
      % --------------------------------------------------------- %
      % Commandability and observability : system without control %
120   % --------------------------------------------------------- %
      disp('%%%%%%%%%  Commandability  %%%%%%%%%%%%%')
      comm=[];
      obs=[];
      for i=1:length(A)
          comm=[comm A^(i-1)*B];
          obs=[obs (A')^(i-1)*C'];
      end
      comm=rank(comm);
      obs=rank(obs);
130
      disp('your commandability rank is')
      disp(comm)
      disp('and must be')
      disp(length(A))

      disp('%%%%%%%%%  Observability  %%%%%%%%%%%%%')
      disp('your observability rank is')
      disp(obs)
      disp('and must be')
140   disp(length(A))

      if (comm==length(A) && obs==length(A))
          disp('Good...')
      end

      %------------------------------------------------------
      %------------------------------------------------------


150   states= {'r1' 'r2' 'y' 'rd1' 'rd2' 'yd'};
      inputs= 'U';
      outputs= {'yd1-yd2'};
      sysCT=ss(A,B,C,D,'statename', states,'inputname', inputs,'outputname', outputs);

      disp('%%%%%%%%% Poles of the open loop system %%%%%%%%')
      disp('%%%%%%%%% Frequency %%%%%%%%%%%')
      f=abs(pole(sysCT)/2/pi)
      f=sort(f);
      f=f(1:2:2*(dim+1));
160   disp('%%%%%%%%%  Damping  %%%%%%%%%%%')
      xi=-real(pole(sysCT))./abs(pole(sysCT))

      % --------------------------------- %
      % Continous to discrete transformation %
      % --------------------------------- %

      fech=input('Which sample frequency ?'); % max 200000kS/s
      % Advice : fe must be 10 to 20 times greater than the observable phenomenom
      % in discrete time system (R. Longchamp)
170   Te=1/fech;

      sysDT=c2d(sysCT,Te);


      % -------------------------------------- %
      % Design of the estimator by Kalman Filter %
      % -------------------------------------- %

      Gnoise=B;
```

```
180    Q=100;         % covariance noise of the system // q/r << quick
       R=1*eye(1);    % covariance noise of the measurement // q/r >> low
       [Lkalm,Pkalm,Zkalm,Ekalm] = dlqe(sysDT.a,Gnoise,sysDT.c,Q,R);
       % Ekalm = poles of the observer
       Lkalm;

       display('%%%% Choice of the regulator %%%%')
       display('1) Pole placement, 2) Linear quadratic regulator');
       choice=0;

190    while((choice<1)||(choice>2))
       choice=input('Choice?')
       end
       % -------------------------------%
       % Pole placement for state feedback %
       % ------------------------------- %

           if(choice==1)
               polereg=createpole(f*2*pi,[0.5 0.02*ones(1,dim)]);
               % taking the same frequency for the pole, but with different damping
200            poleregDT=exp(polereg*Te);
               KregDT=place(sysDT.a,sysDT.b,poleregDT);
               regDT=reg(sysDT,KregDT,Lkalm');
               clDT=feedback(sysDT,regDT,1);
           end

       % ------------- %
       % LQR regulator %
       % ------------- %

210        if(choice==2)
               Q=eye(6);
               R=0.0001; % we are strength on the state, and allow large command if Q/R >>
               [KregDT,Slqg,Elqg]=DLQR(sysDT.a,sysDT.b,Q,R);
               regDT=reg(sysDT,KregDT,Lkalm);
               clDT=feedback(sysDT,regDT,1);
           end

       figure(1)
       pzmap(clDT)
220

       figure(2)
       impulse(sysDT,2);
       hold on
       impulse(clDT,2);

       [Y,t]=impulse(sysDT,5);
       [Ycorr,tcorr]=impulse(clDT,5);

230    figure(3)
       [fr,sg]=TFDr(t,Y(:,1));
       [frcorr,sgcorr]=TFDr(tcorr,Ycorr(:,1));
       subplot(2,1,1)
       semilogx(fr,20*log10(abs(sg)),frcorr,20*log10(abs(sgcorr)))
       xlabel('Frequency [Hz]')
       ylabel('Gain [dB]')
       title('Bode curve of the Velocity at node 55, for a unit impulse at the input')
       grid on
       subplot(2,1,2)
240    %semilogx(fr,atan(imag(sg)./real(sg))*180/pi,fr,atan(imag(sg)./real(sgcorr))*180/pi)
```

Appendix B

Materials

- `B&K` amplifier
- `KEMO` filter
- `PCI-6035E` National Instrument card
- `Velleman K8040` power amplifier