

**uc3m** | Universidad **Carlos III** de Madrid

Bachelor's Degree in Biomedical Engineering  
2017/2018

*Bachelor Thesis*

**Estimation of position and load of  
an SMA muscle by means of  
impedance.**

---

Marta Torralba Álvarez

Tutor

Luis Lorente Moreno  
Leganés, Octubre 2018



*[Incluir en el caso de interés en su publicación en el archivo abierto]*

Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No Comercial - Sin Obra Derivada**



## Abstract

SMA, Shape Memory Alloys, are special kind of materials. As described by the name, this material is made by a metallic alloy, which in this case is a combination of nickel and titanium called nitinol. Due to the various properties SMA has, specially its "memory" capacity, formally called Shape Memory Effect (SME), they can be very useful for many applications. This is why their study has been developed throughout the years, but in order to spread out its use, more progress has to be made in its study to be able to control its behavior.

This work is focused on testing the ability of the impedance response of the material as a measurement method to control the position and the load of an SMA muscle. The idea is to be able to check the movement of the material without the need of using an external position sensor or being able to know the load that it is bearing. So, this Bachelor's thesis will evaluate the feasibility of performing these estimations in relation with the impedance of the material.

In addition, this project offers a view of SMAs, explaining their behavior by understanding their properties and also providing some examples of the different applications this material can be used in.

**Keywords:** SMA, nitinol, smart materials, modeling, position control, load control.



## Acknowledgements

I would like to express my gratitude to all the people who have accompanied me along this adventure.

To my lifelong friends: you have always been there for me. Thank you for all the good times we have had and for helping me get away from my mind from time to time. Especially to you, my friend, the best volleyball co-captain our team could ever have: thank you for always finding a way to cheer me up whenever I'm overwhelmed.

And to my new ones: these four years would not have been the same without you. Thank you for all the moments and getaways full of laughter we have shared along the way.

To my mother, my father and my brother: I cannot thank you enough for everything you do for me. Thank you for always showing me your support and your dedication through good and bad times.

I also need to thank Luis Moreno for offering the opportunity to work in such an amazing project and for all the support and help he gave me through the development of the work.

*"For without hope what do any of us have?"*

- Patrick Rothfuss

(The Wise Man's Fear)



# Contents

<b>Abstract</b>	<b>III</b>
<b>Acknowledgements</b>	<b>V</b>
<b>Contents</b>	<b>IX</b>
<b>List of figures</b>	<b>XI</b>
<b>List of tables</b>	<b>XIII</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Objectives . . . . .	1
1.3 Social and scientific impact . . . . .	2
1.4 Regulatory Framework . . . . .	3
1.5 Contents . . . . .	3
<b>2 State of the Art</b>	<b>5</b>
2.1 Shape Memory Alloys (SMAs) . . . . .	5
2.1.1 Properties and characteristics . . . . .	5
2.1.2 Applications . . . . .	9
2.1.3 Other materials to build muscles . . . . .	13
2.1.4 Electric current to induce change . . . . .	14
<b>3 Materials &amp; Methods</b>	<b>16</b>

3.1	Filtering . . . . .	18
3.2	Modeling: MATLAB implementation . . . . .	21
3.2.1	Estimation method: Differential Evolution . . . . .	22
3.2.2	Input variables . . . . .	26
3.2.3	Output variables . . . . .	28
3.2.4	Screen output . . . . .	29
3.3	Testing the models . . . . .	31
3.3.1	Load . . . . .	32
3.3.2	Position . . . . .	32
<b>4</b>	<b>Results &amp; Discussion</b>	<b>34</b>
4.1	Modeling . . . . .	34
4.2	Load estimation . . . . .	38
4.3	Position estimation . . . . .	45
4.3.1	Dependent on load estimation . . . . .	45
4.3.2	Non dependent on load estimation . . . . .	50
<b>5</b>	<b>Conclusions</b>	<b>55</b>
5.1	Accuracy of modeling . . . . .	55
5.2	Accuracy of load estimation . . . . .	55
5.3	Accuracy of position estimation . . . . .	56
5.4	Future work . . . . .	56
5.4.1	Improving modeling and estimation . . . . .	57



5.4.2	Analysis of heating phase . . . . .	57
5.4.3	Changing electric current . . . . .	58
<b>6</b>	<b>Budget</b>	<b>59</b>
	<b>Annex A: Acronyms</b>	<b>62</b>
	<b>Bibliography</b>	<b>63</b>

## List of Figures

2.1	Stress – strain diagram of the SMA in the superelastic mode. Retrieved from [1]. . . . .	6
2.2	TTR of SMA showing the thermal hysteresis of nitinol. Retrieved from [2]. . . . .	7
2.3	Microscopic view of the transformation between austenite and martensite phases of SMA. The martensite on the right is the twinned martensite and the one on the left the deformed martensite. Retrieved from [3]. . . . .	8
2.4	Geometric appearance of austenite (left) and martensite (right). Distribution of titanium (light gray) and nickel (dark gray) atoms. Retrieved from [4]. . . . .	9
2.5	Stent made of SMA. Retrieved from [5]. . . . .	11
2.6	Robotic hand created with SMA. Retrieved from [6]. . . . .	12
2.7	Orthodontic wire made of SMA. Retrieved from [5]. . . . .	12
2.8	SMA used for self-deployable antennas. (Courtesy Goodyear Aerospace Corporation). Retrieved from [7]. . . . .	13
2.9	DSC curve of a SMA showing the necessary heat flow. Retrieved from [8]. . . . .	15
3.1	Impedance signal of 100 g of load. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)). . . . .	17
3.2	Impedance signal of 100 g of load between cooling time and final relaxation time. . . . .	18
3.3	Filtered impedance signal of 100 g of load. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)). . . . .	19

3.4	Filtered impedance signal of 100 g of load between cooling time and final relaxation time. . . . .	20
3.5	Filtered and not filtered impedance signal of 100 g of load. . . . .	21
3.6	Differential Evolution Algorithm. . . . .	24
3.7	Position signal of a load of 100 g. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)). . .	30
3.8	Position signal of 100 g of load between cooling time and final relaxation time. . . . .	31
4.1	Variables printed on the screen during optimization. . . . .	35
4.2	Estimation models. . . . .	36
4.3	Estimation models for all loads. . . . .	37
4.4	Impedances for load estimation at the 4 different times stated. . . .	39
4.5	Load estimation at 4 different times showing the real load (in yellow) and the two estimated loads (in blue and red). . . . .	41
4.6	Mean impedance during relaxation of the different loads studied. . .	42
4.7	Variance of the impedance during relaxation of the different loads studied. . . . .	43
4.8	Maximum and minimum values of the impedance of the different loads studied. . . . .	44
4.9	Difference between maximum and minimum values of impedance of the different loads. . . . .	45
4.10	Position estimation at the 4 different times stated. . . . .	50
4.11	Position estimation at the 4 different times stated. . . . .	54

## Table Index

2.1	Characteristic Lattice Parameters (Length in Angstrom (a, b, c) and Angle in degrees (ab, bc, ac)) for Ni–Ti Alloys as seen in Figure 2.4. Retrieved from [4]. . . . .	9
4.1	Input parameters of the estimation function. . . . .	34
4.2	Sum of Squared Differences between model and signal. . . . .	38
4.3	Load estimation for the 4 different times stated. Highlighted in blue are the real positions that belong to the range of the estimated position. . . . .	40
4.4	Position estimation at time $t = 4408$ ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions. .	46
4.5	Position estimation at time $t = 6908$ ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions. .	47
4.6	Position estimation at time $t = 9408$ ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions. .	48
4.7	Position estimation at time $t = 11908$ ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions. .	49
4.8	Position estimation at time $t = 4408$ ms. Also shown the value of SSD in the third column. . . . .	51
4.9	Position estimation at time $t = 6908$ ms. Also shown the value of SSD in the third column. . . . .	52
4.10	Position estimation at time $t = 9408$ ms. Also shown the value of SSD in the third column. . . . .	52
4.11	Position estimation at time $t = 11908$ ms. Also shown the value of SSD in the third column. . . . .	53

6.1	Personnel budget. . . . .	59
6.2	Materials budget. . . . .	59
6.3	Equipment budget. . . . .	60
6.4	Software budget. . . . .	60
6.5	Total cost. . . . .	61



# 1 Introduction

The motivation and objectives that are to be achieved by the end of this project will be discussed under this section. Here, it can also be found the social context the problem refers to and the regulatory framework as well as the outline of this work. The outline of the work will also be presented adding a brief description of what can be found within each chapter.

## 1.1 Motivation

The investigation and development of smart materials that possess many important properties, are key for the progress in the different application fields they can be used on, which, are many, as will be discussed. Therefore, it is interesting to solve small problems or steps in the most practical way, that is, cheap and efficient, to improve their applications.

In this particular case, the material under study is a Shape Memory Alloy (SMA) used to mimic the function of muscles. If this material is going to perform the same functions as a muscle, it should be able to bear different loads, and it needs to have the ability to contract and relax, that is, be able to change its position. This is why the project is focused on estimating this variables (load and position) without using an external sensor that may make the material more expensive to use. In the case that the estimation is performed correctly, the sensor would be substituted by the measurement of the impedance response of the material when subjected to an external current (external stimulus like in the case of muscles) that produces a change in its phase.

## 1.2 Objectives

The main objective of the project is to propose some models using four given samples of impedance and position data of SMA bearing 17 different loads, so that using them, one could be able to find out at any time the load and position

of a given SMA impedance signal without the need of using a sensor specific for that purpose.

So, the goal can be broken down into three main steps:

1. Propose the right model of the impedance responses belonging to the different loads.
2. Using these models estimate the load of another given SMA impedance signal.
3. Lastly, knowing the load the impedance signal belongs to, estimate the position of the SMA muscle.

### **1.3 Social and scientific impact**

The benefits that can be obtained from this project, both in the scientific and in the social fields, are enormous.

By increasing the knowledge about this material, that is, improving the control over its behavior, it will be improving its great application capacity. This could be translated into an advancement in the social field because by making progress or developing new applications could improve the quality of life in different ways. Applications in the field of medicine are a direct example of how the study of SMAs could have an impact in society, by developing new ways to overcome medical problems. For example, the material could be used to design new orthosis<sup>1</sup> are being developed [9], which would allow for a better recovery of the limb; or using it as muscles in robotic hands [10].

In the scientific field, the continuous study and development of smart materials is very important since they could help improve the designs that are being used currently in their different applications. By investigating other methods to estimate the position and the load of an SMA wire, more knowledge about the control of the material could be obtained and it would allow its use in an efficient

---

<sup>1</sup>Orthopedic apparatus used to support, align, prevent, or correct deformities or to improve function of movable parts of the body [9].



way needing less equipment for the measurements so that its use could be spread out in the different areas.

## 1.4 Regulatory Framework

For the commercialization of different applications in the field of medicine in Europe, several regulatory approvals must be obtained . It has to satisfy the regulations for medical devices, like the Council Directive 93/42/EEC on Medical Devices. In Spain, they have to comply with the Real Decreto 1591/2009 and they have to be approved by the AEMPS (Agencia Española de Medicamentos y Productos Sanitarios) [11].

## 1.5 Contents

This Bachelor Thesis is divided into 6 different chapters:

- **Chapter 1: Introduction**

In the first chapter, the introduction, the objectives and motivation for the project will be addressed to understand the importance of the work that is being done and what it is wanted to accomplish by the end of it. It will also offer a brief view of the impacts that the development of SMAs could have over science and society and the regulations it must follow.

- **Chapter 2: State of the art**

This second chapter, takes a closer look into the material being analyzed in the project: Shape Memory Alloys, understanding why it is being used understanding its most important properties and characteristics and what are its current applications. It will also take a look into other materials that are being used to build muscles.

- **Chapter 3: Materials & methods**

This chapter explains the materials or data that were needed for the development of the project as well as the description of the different steps

taken in order to reach the objective and final solution. This will include the design of the software.

- **Chapter 4: Results**

In this chapter the results of the tests explained in Chapter 3 are shown and discussed.

- **Chapter 5: Conclusions**

This chapter contains the conclusions achieved by the developing of the project analyzing the objectives that were set and the results obtained. It also includes some future paths that could be followed to further develop this analysis.

- **Chapter 6: Budget**

In this last chapter, the costs associated to the project are broken down.

## 2 State of the Art

Under this section a review of the characteristics and uses of the material employed in this project will be explained to understand its current applications. Also, some other materials that are being used to recreate muscles will be presented.

### 2.1 Shape Memory Alloys (SMAs)

The material under study in this project is the SMA, which stands for Shape Memory Alloy, a thermo-shrinkable material made of nitinol: a combination of nickel and titanium, with special properties that make it useful in the robotics field (used as muscles for example) and also for actuators.

#### 2.1.1 Properties and characteristics

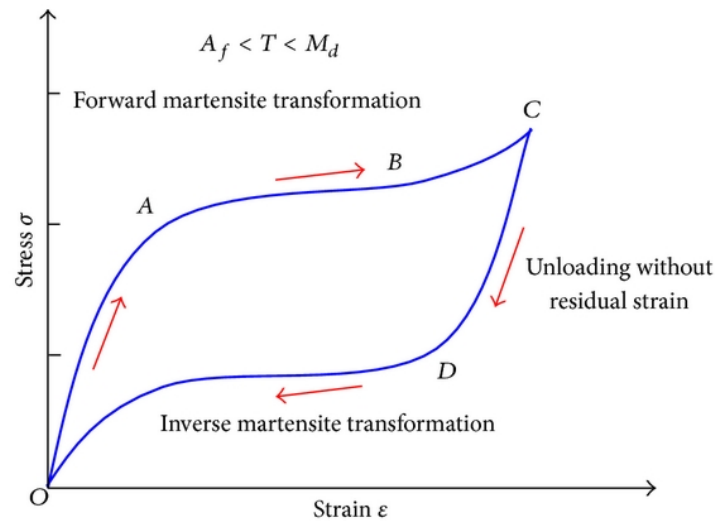
This material has an unique and very important characteristic, its “memory”, which is also referred as Shape Memory Effect (SME) and it correlates with two other important characteristics: superelasticity and thermoelasticity. Due to the Shape Memory Effect, the material has the ability to return to its original shape after being subjected to an external stimulus, like being exposed to a heat source. This means that it is capable of undergoing serious deformations without them being permanently irreversible. The properties of superelasticity and thermoelasticity are closely related to the shape memory effect in the way that as the material has that great capacity of elasticity, it is more difficult to reach the point of irreversible deformation. The first one, which can also be referred as pseudoelasticity, allows the alloy to exhibit a comparatively large recoverable strain, a reversible strain, caused by the phase transformation between austenitic and martensitic phases of the crystal. And the second one, thermoelasticity, indicates the functional dependence of the recoverable strain on thermal conditions.

As it has been mentioned, the property of superelasticity is highly related to the shape memory effect, as it means that the wire can be reversibly deformed up

to high values of strain, up to 7-8%, which is almost forty times the capacity of stainless steel [12].

The change between the two phases of the material (austenitic and martensitic) can also be produced by a variation of stress, that is called the superelastic behavior. As it can be seen in Figure 2.1, the application of a shear stress to the material induces in it a change from the austenitic phase to the martensitic one (forward transformation). After this transformation and while the stress is maintained, the alloy remains deformed in the martensitic configuration; but, as soon as the shear stress is removed, the martensite becomes unstable and the phase is reverted back to austenite (reverse transformation) [12].

As it can be seen in the figure, this variation of stress and the change in the phase, also cause an alteration in the strain of the material. These large variations of strain are what make the material superelastic, as it is able to bear high deformations of the strain (around 7-8%) and continue being reversible returning to its original shape after unloading.

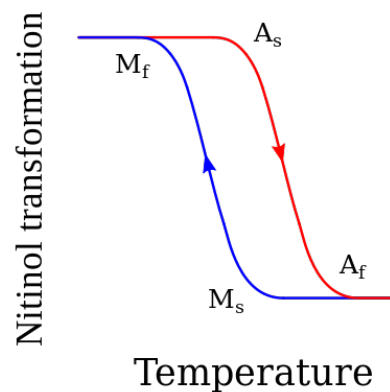


**Figure 2.1:** Stress – strain diagram of the SMA in the superelastic mode. Retrieved from [1].

Thermoelasticity is the other characteristic that is related to the SME, and it applies to the temperature behavior of SMA. Each alloy has a specific temperature range in which the transition between phases occur, the TTR (Transformation Temperature). These temperatures are  $M_s$  and  $M_f$ , initial and final at which

martensite is formed; and  $A_s$  and  $A_f$  which are the corresponding temperatures at which the austenitic phase is created. This temperature range is shown in Figure 2.2. It can be seen that at lower temperatures the material is stable in its martensitic phase, and the transformation begins when the temperature increases from  $M_s$  to  $A_f$ , where the alloy changes to the austenitic phase [13].

As it has been mentioned, the material does not change its phase in a specific temperature, but it does so in a range of temperatures called TTR that has a typical value of (20-50 °C) [2], although it is really dependent on the type of alloy. The transformation begins either at  $A_s$  or  $M_s$ , depending if the material is being heated or cooled, and it ends in  $A_f$  and  $M_f$  respectively. The first ones ( $A_s$  and  $A_f$ ) represent the transformation from martensite to austenite and the second ones ( $M_s$  and  $M_f$ ) the opposite transformation. This hysteresis <sup>2</sup> of the material can be seen in Figure 2.2. In this process, some of the mechanical energy is lost.



**Figure 2.2:** TTR of SMA showing the thermal hysteresis of nitinol. *Retrieved from [2].*

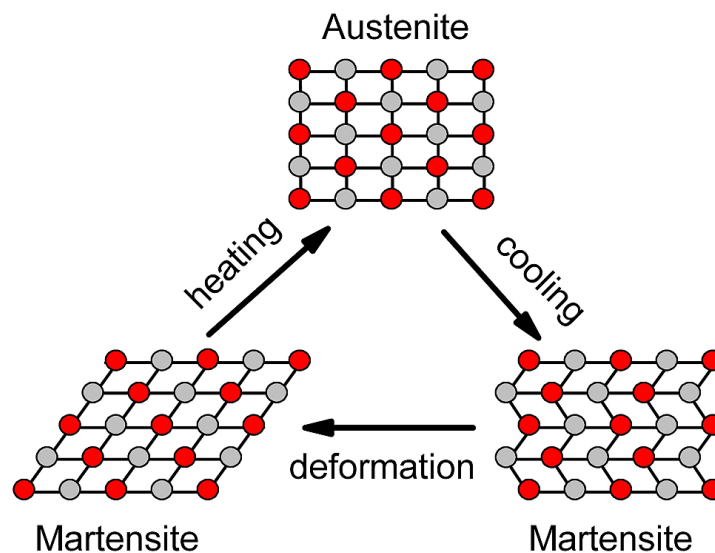
As mentioned, these properties of the material are due to the characteristics of the transformation between austenitic and martensitic phases of the crystal: the reorganization that the crystal structure undergoes during this phase shift. These two phases of the crystal correspond to the stable structure of the material at either low (martensitic phase) or high (austenitic phase) temperatures.

The first one, martensitic, is a fragile phase that can be easily deformed. It is characterized by an off-centered structure and highly crystalline phase (it has

<sup>2</sup>Lag in response exhibited by a body in reacting to changes in the forces [14]. In this case retardation in the transformation upon heating.

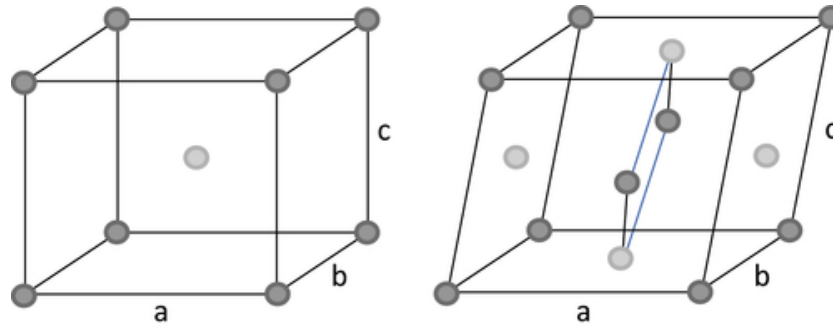
a monoclinic crystal structure) that can find its stability in two different forms as shown in Figure 2.3 (twinned and detwinned). The geometry of the austenitic phase, which corresponds to the stable phase of the material at high temperatures, is a centered cubic geometry, so it is more symmetric than the martensitic one. The reorganization of nickel and titanium atoms in each of the phases geometry can be seen in Figure 2.4 [4].

So the transformation between the phases can be obtained by changing the temperature, either cooling or heating, as it can be seen in Figure 2.3. The martensite view of the right shows the molecular structure of martensite, which is twinned, and after deformation it changes its shape to the one on the left side (the deformed detwinned martensite).



**Figure 2.3:** Microscopic view of the transformation between austenite and martensite phases of SMA. The martensite on the right is the twinned martensite and the one on the left the deformed martensite. Retrieved from [3].

The lengths of the sides ( $a$ ,  $b$  and  $c$ ) and the angles between them ( $ab$ ,  $bc$ ,  $ac$ ) of the unit cell in the crystal lattices of each of the structures is noted in Table 2.1, where it can be seen what was previously mentioned: for the martensitic phase it can be observed that the length of the sides are not equal and that the angle between  $a$  and  $c$  is not a square angle, whereas in the case of the austenitic phase the structure of the crystal is a perfect cube.



**Figure 2.4:** Geometric appearance of austenite (left) and martensite (right). Distribution of titanium (light gray) and nickel (dark gray) atoms. Retrieved from [4].

Lattice Parameters	Length (Å)			Angle (deg)		
	a	b	c	ab	bc	ac
Martensite	2.9	4.1	4.7	90	90	98
Austenite	3.0	3.0	3.0	90	90	90

**Table 2.1:** Characteristic Lattice Parameters (Length in Angstrom (a, b, c) and Angle in degrees (ab, bc, ac)) for Ni–Ti Alloys as seen in Figure 2.4. Retrieved from [4].

SMA also has many other interesting mechanical characteristics; like, for example, when it is heated it contracts, showing a complete opposite behavior when it is compared to all other metals, that expand when they are subjected to higher temperatures; moreover, the thermal movement it produces, the movement of contraction and extension, is a hundred times higher than the one obtained with standard metals [3].

### 2.1.2 Applications

As it can be deduced by the name of the material, the shape memory effect is the most important characteristic that makes this material suitable for many applications under different fields of study: medicine, robotics, construction, etc. However, it possesses two other important properties that make it suitable for many applications are its resistance to corrosion and its non-toxicity, thus allowing its use in the human body.

The main use of SMAs is as actuators, being the component responsible of moving or controlling the system by "changing" its mechanical characteristics in response to temperature. As an actuator they can provide force or displacement in the selected direction [15].

- Medicine

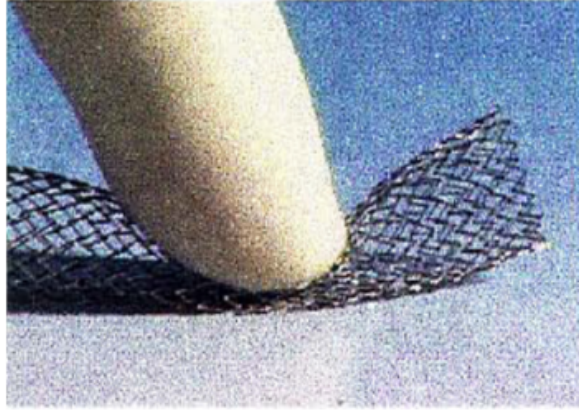
In addition to the shape memory effect, in the field of medicine, biocompatibility plays a very important role so that it can be used for applications for the human body without it reacting against the material.

The behavior of this material, its thermal movement of contraction and extension that was previously mentioned, resembles the operation principle of a muscle: it contracts when heated (when subjected to an external stimuli), and then it relaxes returning to its original shape. This movement of the material together with the memory effect make it suitable for one of the main applications in the field of robotics and medicine: the use of SMAs as muscles. Some other applications for Shape Memory Alloys in the field of medicine together with the robotics field have been developed making use of the important properties that the material presents. This is the case of robotic hands or other joints for the human body that can also take advantage of the memory effect and thermal movement like muscles.

Among other applications in this area is the use of SMAs to replace functions of some organs in the human body. This can be the case of the heart or the kidneys, where this material is used as a micro pump.

They are also used to build devices that later could be needed in non-invasive surgery. The advantage that this presents with respect to normally used tools, is that the device built with SMA can be deformed when subjected to an external stimulus and when this is removed, the tool will go back to its original shape, enabling its use in devices that can reinstate the flow in narrow places such as stents, that are used to decongest clogged arteries and to prevent the collapse of blood vessels.





**Figure 2.5:** Stent made of SMA. Retrieved from [5].

Applications in the neuroscience field are also being developed. There are some works where the SMA is applied in the field of neuromuscular rehabilitation, like being used to design orthosis for spastic limb repositioning, which may prevent from disuse and immobility of the limb by allowing residual voluntary movement of the limb, as proposed by [16]. They can be also used for the mobilization of paralyzed hands to assist in hand rehabilitation like the method proposed by [17].

- Robotics

Besides the previous application belonging to the field of medicine and robotics, Shape Memory Alloys are also used in the robotics field as actuators.

An example of this can be seen in robotic hands which have the advantage that can be used for many purposes; a multi-finger robotic hand can serve as an end effector element as it possesses the ability to grip different objects, being able to perform many industrial operations like transferring, assembling, welding and painting. As it has been said, the SMA is used in this cases as an actuator. So, SMA in this case is responsible for moving the hand to specific positions at specific temperatures, and it also takes advantage of the similarity to human muscle structures that was previously stated allowing for a more flexible movement [10].



**Figure 2.6:** Robotic hand created with SMA. Retrieved from [6].

- Dentistry

This alloy has also been used in the field of dentistry to manufacture orthodontic wires due to its good mechanical properties, especially, the properties that make this a suitable material for this application are its shape memory property, superelasticity, high ductility, resistance to corrosion and biocompatibility [12].



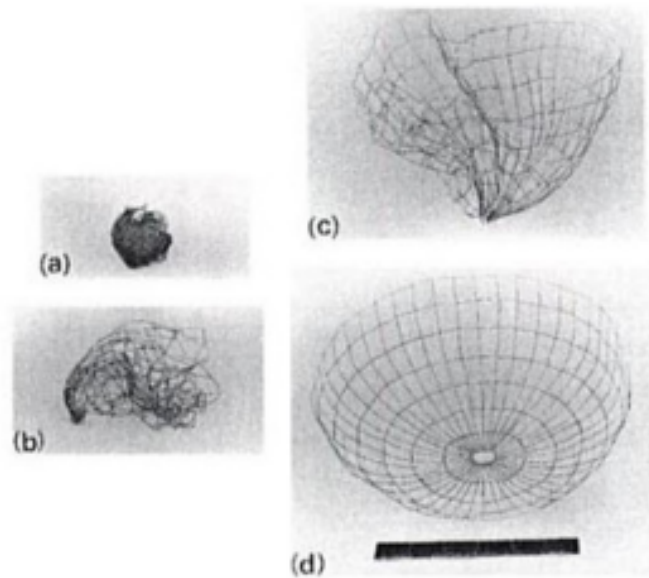
**Figure 2.7:** Orthodontic wire made of SMA. Retrieved from [5].

- Construction

As it has been mentioned, the field covered by SMA is very wide, having also other applications like being used in the civil engineering field, making use of large SMA wires as damping elements in bridges and in seismic resisting systems. The reason they are used for these purposes is because of their excellent energy dissipation and re-centering capabilities that come from the great damping capacity they possess and the ability to deliver high plateau strains at almost constant stresses as seen in Figure 2.1 [18].

- Others

They can also be used for aerospace applications, like is the case of building flexible antenna systems that could be reconfigured in space [7]. To impose a permanent shape on the antenna some heat treatment is required. Then, to send it to space, it is cooled down to its martensitic phase where it can be easily deformed and occupy less space (Figure 2.8 (a)). Once it is launched, the solar heating will be responsible of raising the temperature above the reverse transformation temperature of the system so that the permanent shape is recovered (Figure 2.8 (b-d)) [19].



**Figure 2.8:** SMA used for self-deployable antennas. (Courtesy Goodyear Aerospace Corporation). Retrieved from [7].

### 2.1.3 Other materials to build muscles

Other materials that can be used to build muscles include polymers. These materials resemble muscles in their ability to reversibly expand and contract in length and volume, some of the materials that fall under this category are elastomers, conducting polymers, ionically conducting polymers, and carbon nanotubes [20]. Two groups of polymers that can be used for artificial muscles will be discussed, which are Electroactive Polymers (EAPs) and Shape Memory Polymers (SMPs).

- EAPs

Inside this group, two major classes of polymers that may be used are found: electric Electroactive Polymers and ionic Electroactive Polymers. The first group needs an electric field in order to experiment some change and the actuation in the polymers from the second comes from the presence and movement of ions in the electrolyte phase [20].

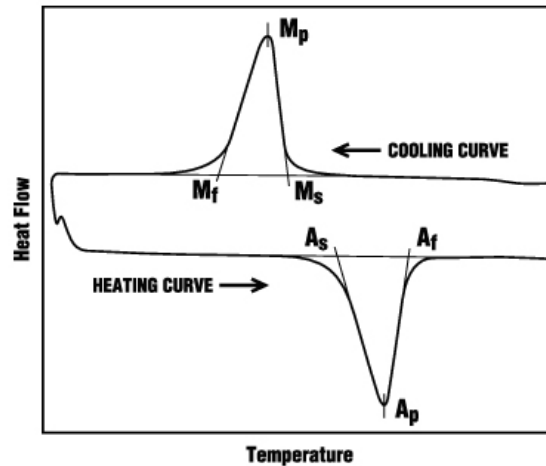
- SMPs

The second group of polymers mentioned are the ones called SMPs (Shape Memory Polymers). They have the ability of being deformed upon the application of a small load when heated above their glass transition temperature ( $T_g$ ), and being able to maintain that shape in a rigid form once they have been cooled below  $T_g$ . If they are heated above  $T_g$  again, they will return to their predefined shape. So, its shape can be changed by increasing its temperature, deforming it and then cooling it down fixing the structure in a rigid, actuated state that can be maintained without the need of any control system [21].

#### **2.1.4 Electric current to induce change**

To activate the material, an electric current can be used, either by applying electrical currents to circulate through the SMA, which will heat up the material raising its temperature; or by external heating which can be either convection, radiation or conduction [22].

The heating the material needs depends on the conditions it is under. These can be observed in the graphics obtained by performing Differential Scanning Calorimeter (DSC) experiments in which these heat flow is plotted. Figure 2.9 shows both the heating process and the cooling one.



**Figure 2.9:** DSC curve of a SMA showing the necessary heat flow. Retrieved from [8].

- Heating curve

A constant heat flow is applied resulting in a temperature increase up to  $A_s$ , at which point, the transformation to austenitic phase will begin and it will be necessary to increase the heat flow applied to the SMA. When most of the martensite has been transformed, this heat flow will decrease until becoming constant again from  $A_f$ , time when the only existing phase will be austenite.

- Cooling curve

The same thing happens as in the previous case, starting with constant flow but in this case, when reaching  $M_s$ , the material starts to give up the energy obtained from the heating, so the heat flow decreases until most of the austenite has been transformed into martensite, and then it becomes constant again from  $M_f$  forward.

### 3 Materials & Methods

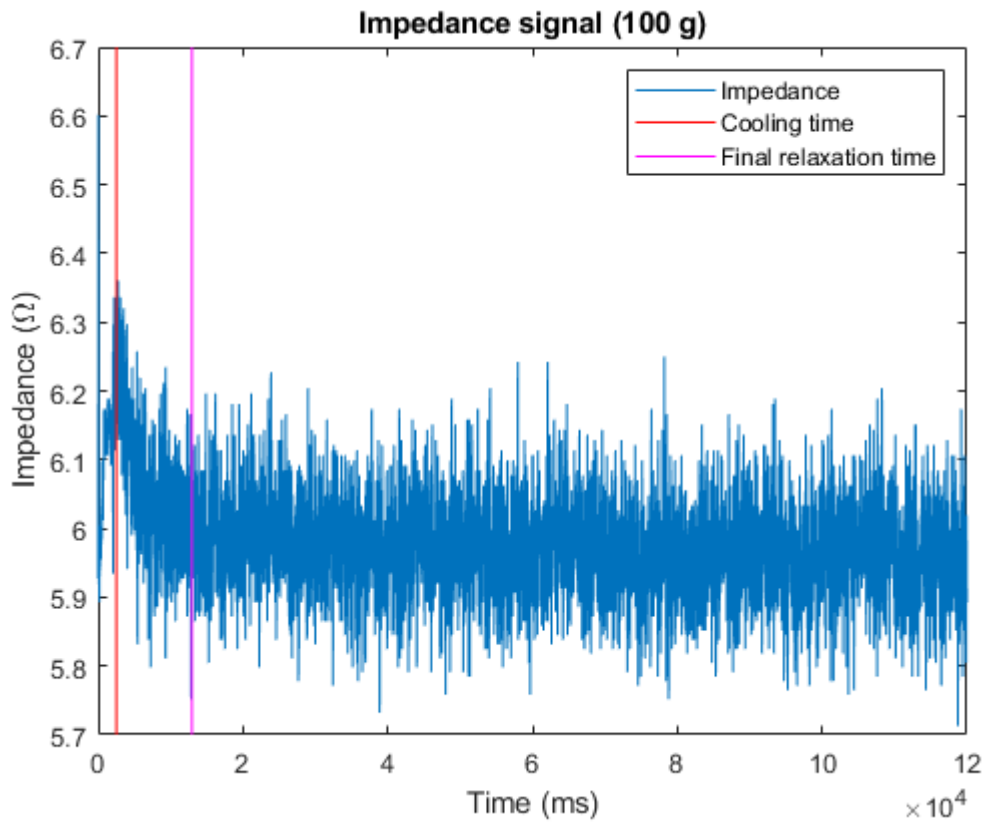
To find the correlation between the impedance signal and the position signal of the different loads, as well as the relation between impedance values at specific time points and the load of SMA, it is necessary to work with the impedance responses of an SMA wire with different loads to be able to create a model of each of them that may be used to differentiate between loads and estimate them. The software that is used to analyze all the data is MATLAB.

The data used contains four samples for each of the 17 different loads (ranging from 100 g to 3000 g), each of them having the position and impedance signals. Data values for each sample were taken every 1 ms, and the values of loads used are the following ones: 100 g, 200 g, 300 g, 400 g, 500 g, 600 g, 700 g, 800 g, 900 g, 1000 g, 1100 g, 1250 g, 1500 g, 1750 g, 2000 g, 2500 g and 3000 g.

Three out of the four samples of each load were used for the modeling part and the remaining one was the one used to test the work. The reason for using three signals for the modeling was to calculate their mean value to try and reduce the errors before working on them. So, the impedance signal that was filtered and modeled was the result of calculating the mean of the three samples.

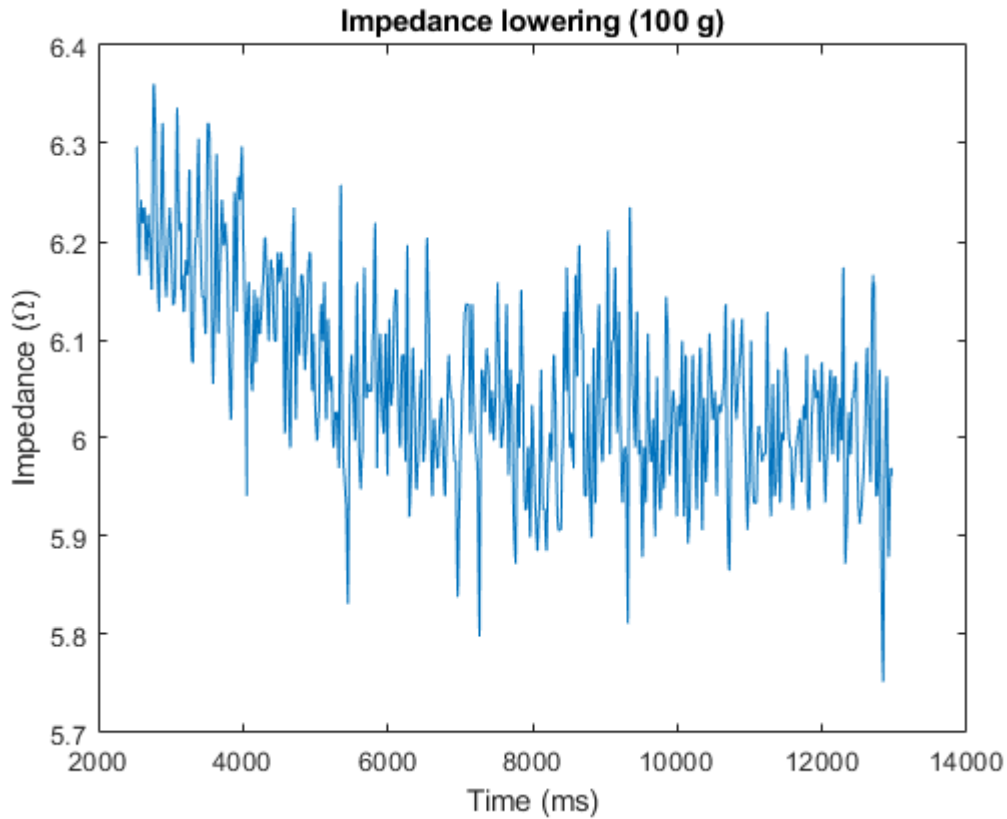
For the position a model using an optimization problem did not have to be created, just computing the mean of three position signals was enough to serve as a model for each load.

Figure 3.1 shows one of the impedance signals available in the data, in this case corresponding to a load of 100 g.



**Figure 3.1:** Impedance signal of 100 g of load. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)).

The modeling of the signal will be made for the cooling phase, it is an analysis of the impedance lowering. It will go from  $t = 2533$  ms that corresponds to cooling time until final relaxation time, which is  $t = 12970$  ms.



**Figure 3.2:** Impedance signal of 100 g of load between cooling time and final relaxation time.

### 3.1 Filtering

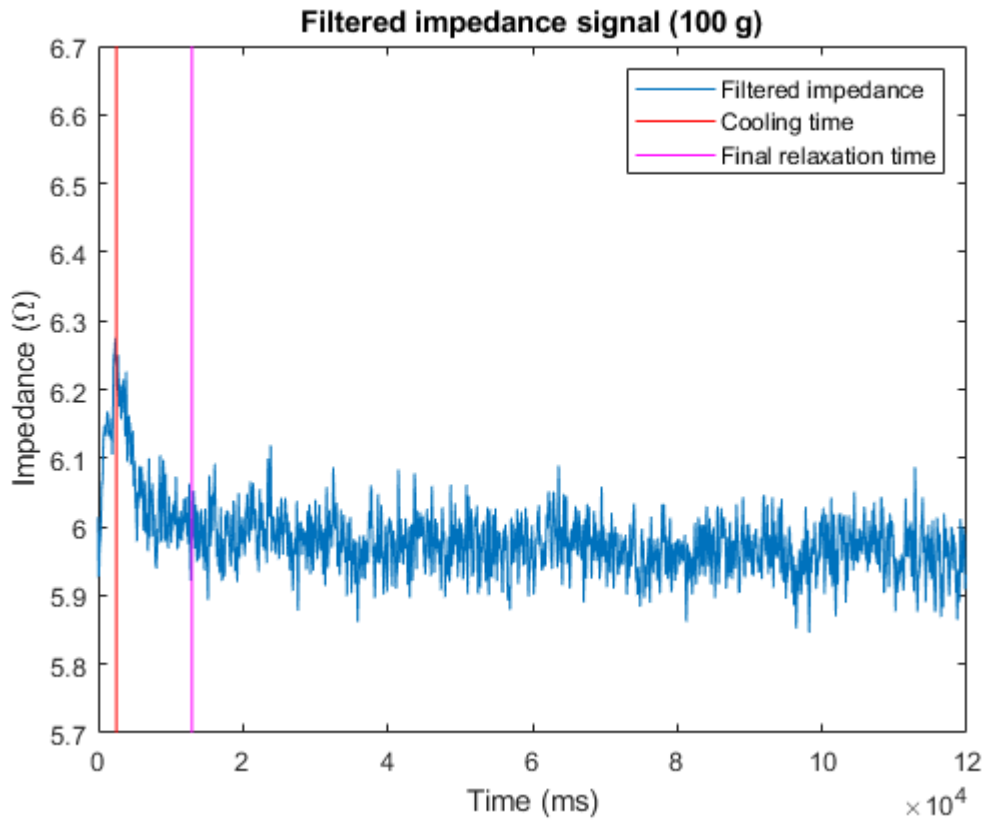
The first step to treat the data to be able to model the impedance responses is to apply a filter that will eliminate the outliers of the impedance signal that will affect the modeling.

The filter used for this purpose, needs to be a one-dimensional smoothing filter to eliminate the noisy data. The one chosen is a moving-average filter. It computes the average of the data values over an specified window that is moved along the data. It follows equation 3.1.

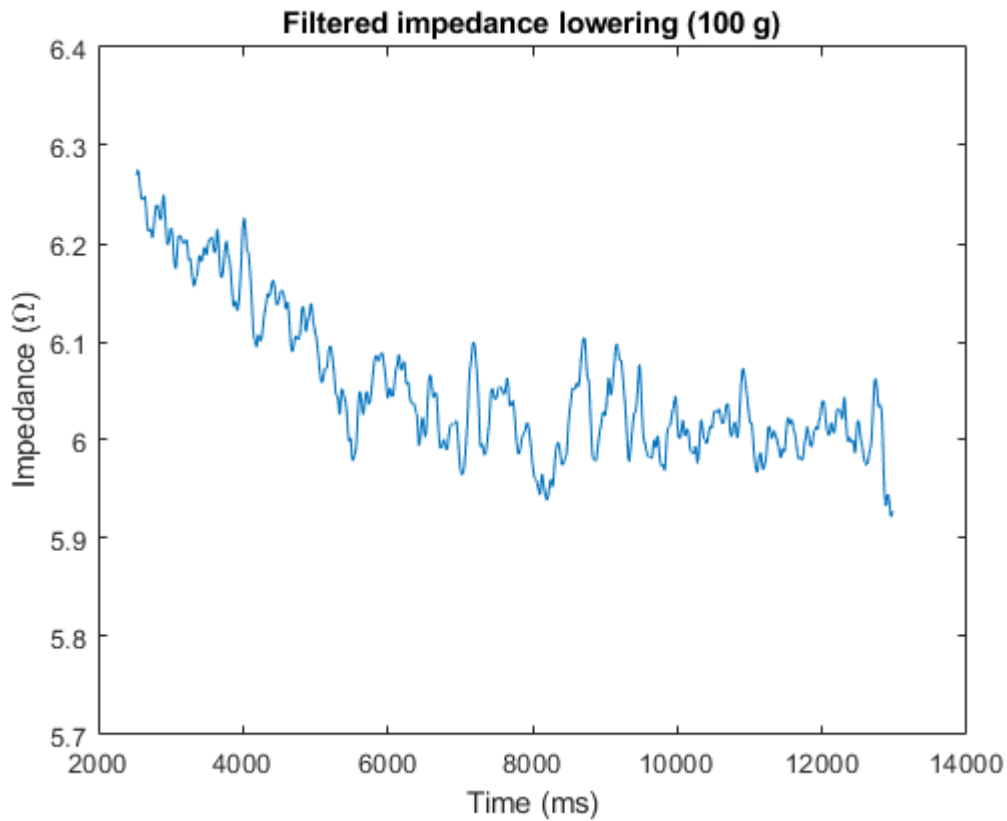
$$y(n) = \frac{1}{windowSize} (x(n) + x(n-1) + \dots + x(n - (windowSize - 1))) \quad (3.1)$$



The window size used to filter the impedance responses was 150 as the amount of data contained in each of the signal files was big (there were 120002 values for each of them). So, in this case, the filtered signal will be obtained by making averages of 150 values while moving the window along the data.



**Figure 3.3:** Filtered impedance signal of 100 g of load. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)).



**Figure 3.4:** Filtered impedance signal of 100 g of load between cooling time and final relaxation time.

In Figure 3.5, both the initial impedance signal and the resulting signal after the filtering has been performed are plotted so that the reduction of the outliers can be observed. In the first graph of the figure, the data is shown over the whole range of time, and the second graph is focused on the lowering part of the impedance where the analysis will be performed. That is, between the cooling time and the final relaxation time.

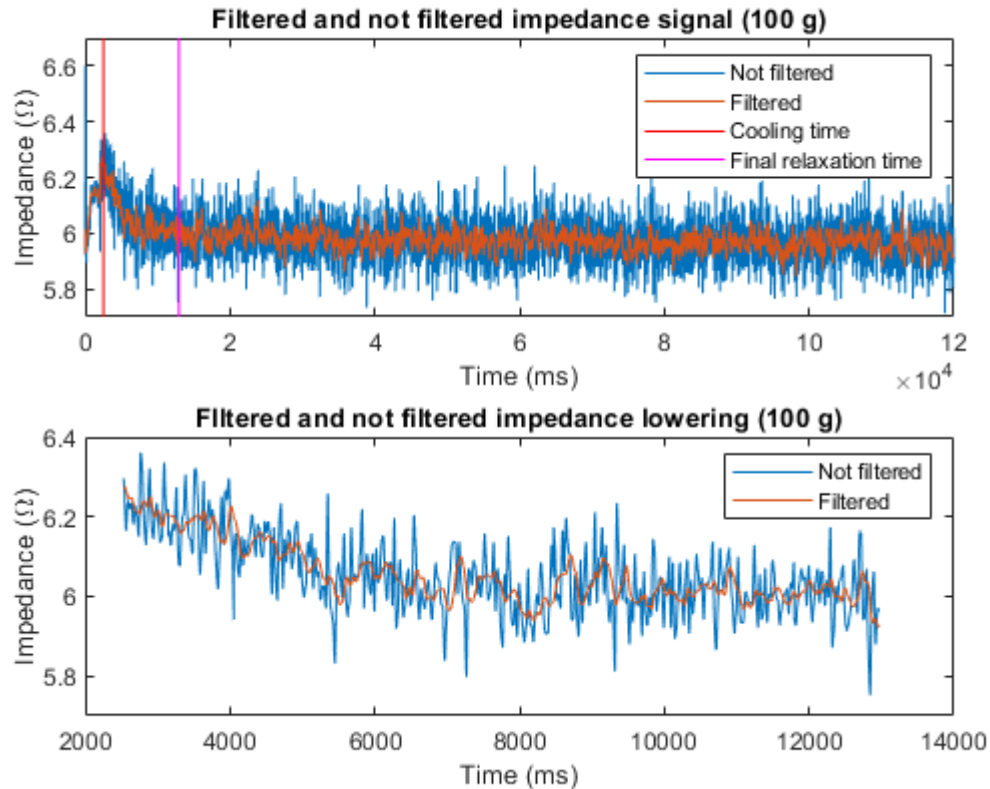


Figure 3.5: Filtered and not filtered impedance signal of 100 g of load.

### 3.2 Modeling: MATLAB implementation

The approach taken to solve the problem of optimization was to use the DE minimization method (differential evolution minimization), which is a simple population based, stochastic function minimizer [23] that belongs to the group of Evolutionary Algorithms (EA). This kind of algorithms are based on imitating biological evolution to solve the optimization problem by following four actions: initialization, selection, genetic operators, and termination [24]. In the case of DE, the method used to guide the search process is by using the distance and direction information from the current population [25].

This algorithm is appropriate for this optimization as it is used for problems over continuous domains, which corresponds to this case, offering some advantages such as a simple structure that makes it easy to use, as well as its speed

and robustness. It uses an iterative approach to reach the optimal solution by improving the candidates [26].

As it has been previously mentioned, the modeling is performed on the lowering part of the impedance (extension phase), covering the range from cooling time until final relaxation time. So, the signal that is used as an input for the model is the one shown in Figure 3.4 (taking into account the data between the selected times of the filtered impedance signal)

The model for the signal is made by testing different estimation methods. First of all, the signal is down-sampled, a sample is taken every 25 ms; and the cooling time and the final relaxation time are selected so that the lowering analysis can be done between those times.

### 3.2.1 Estimation method: Differential Evolution

As mentioned, the Differential Evolution algorithm is a population based search, so it starts by generating a random population to use as the initial one. Then, three random vectors are selected from the population so that the genetic operators can be applied. One of these vectors will be the target vector and the other two will be used to build the weighted difference vector. The genetic operators applied in the DE algorithm are mutation and crossover.

- Mutation

Each individual (parent vector  $(x_i(t))$ ) of the current population will be mutated to generate a trial vector  $(u_i(t))$  using a target vector  $(x_{i1}(t))$  and a weighted difference obtained by using two random vectors  $(x_{i2}(t)$  and  $x_{i3}(t))$  where  $i \neq i1 \neq i2 \neq i3$ , as seen in equation 3.2 [25].

$$u_i(t) = x_{i1}(t) + F(x_{i2}(t) - x_{i3}(t)) \quad (3.2)$$

where F, as it will be explained after, corresponds to the weighted factor that will control the amplification of the differential variation [25].

It can be seen that the mutation step size is directly influenced by the difference between individuals from the current population, which means that the distances between individuals are considered in the search process;

over time, these distances will become smaller until all individuals converge to the same solution [25].

- Crossover

After the mutation operator is applied, a discrete recombination of the trial vector ( $u_i(t)$ ) and the parent vector ( $x_i(t)$ ) will be produced to generate offspring ( $x'_i(t)$ ). Equation 3.3 shows the implementation of this operator [25].

$$x'_{ij}(t) = \begin{cases} u_{ij}(t), & \text{if } j \in J \\ x_{ij}(t), & \text{otherwise} \end{cases} \quad (3.3)$$

where  $J$  is the set of indices that will undergo perturbation (set of crossover points) [25].

The approach used to determine  $J$  in this project is the binomial crossover, so, the set of points that will belong to  $J$  are randomly selected by considering the probability (CR) of the points being included. A higher probability means more points will be included in the set of crossover points  $J$  than if a lower CR would have been selected [25].

After the crossover operator is applied, deterministic selection takes place: if the offspring provides a better fitness than the parent, then it will replace it in the next generation; otherwise, the parent will remain in the population.

A good thing about this algorithm is that it does not use the gradient of the signal to be optimized, so it can work well for noisy problems such as this case. The reason why it does not need the gradient is because the optimization of the problem is performed by creating new candidate solutions as a combination of existing ones and then keeping the ones that return the best score of the optimization [27].

The diagram shown in Figure 3.2.1 shows the different steps followed by the algorithm to reach the final solution.

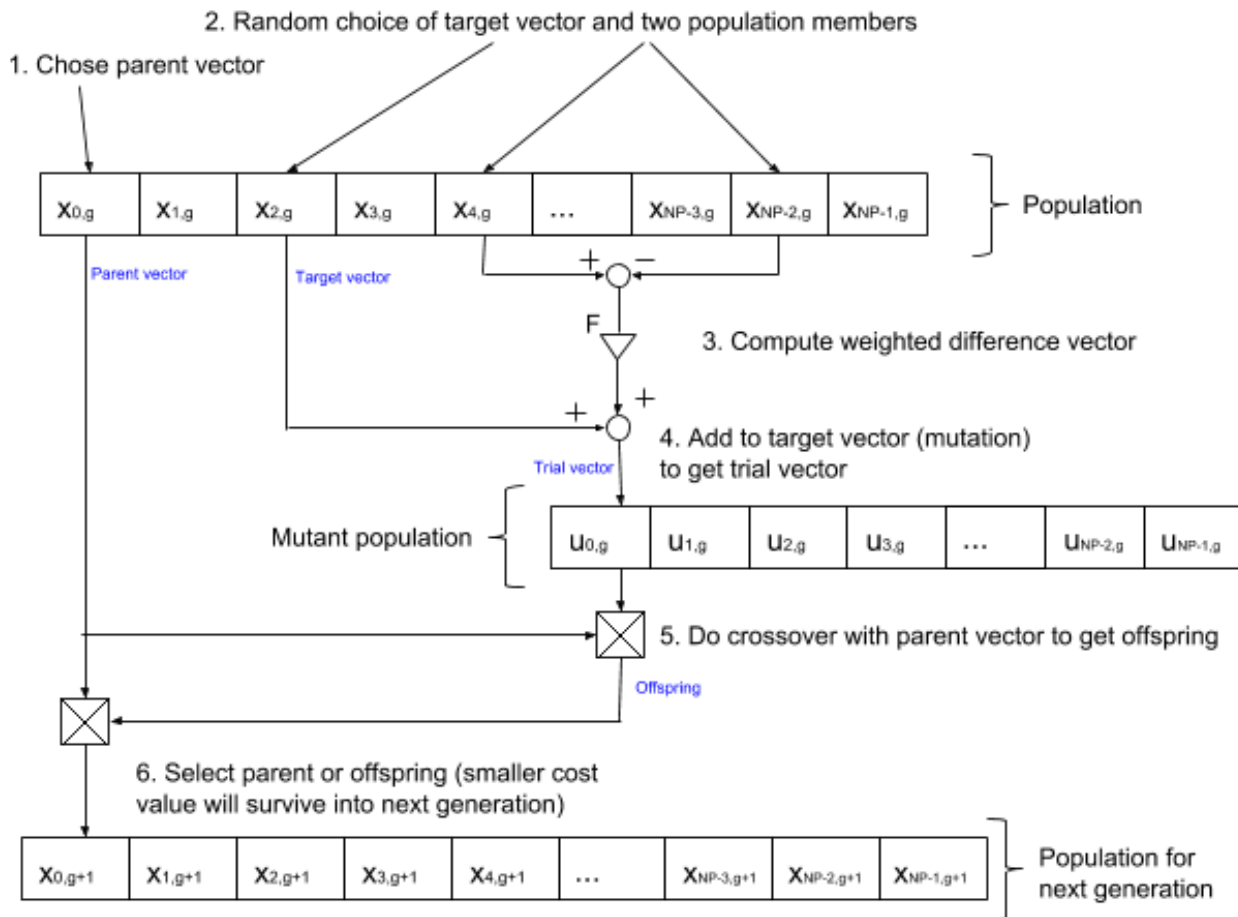


Figure 3.6: Differential Evolution Algorithm.

## Objective function

The program is aimed towards calculating a model for our signal, so the objective function, also called cost function, is the equation that needs to be optimized given certain constraints.

It will be minimized using nonlinear programming techniques. It needs to be minimized because it will represent the difference between our data and the optimized data, which has to be minimum. To do so, several cost functions were tried to find the one that best describes the problem. This different objective functions end up giving different results so it is possible to chose one. The input variables selected to find the right optimization mechanism will be discussed in the following subsection.

The different cost functions that were tried (four types of exponential functions,

a Gaussian, a linear, a cubic polynomial and a sinusoidal one) are shown in equations 3.4 to 3.11.

$a, b, c, d, e$  are the parameters that are being calculated to construct the model performing the estimation. They correspond to the different positions (a = position 1, b = position 2...) of the vector returned by the estimation function (**bestmem**). The number of parameters used for each cost function differs, ones need more and others less, this variable (D) is selected by the user, as it will be discussed later on.

- Exponential

$$y = a - b * \exp(-ii/c-d) \quad (3.4)$$

- Growing Linear Exponential

$$y = b + c * TC + d * TC^2 + e * \exp(TC) \quad (3.5)$$

- Linear Exponential

$$y = b + c * TC + d * TC^2 + e * \exp(-TC) \quad (3.6)$$

- Linear Exponential 2

$$y = b + c * TC + d * \exp(-TC^2) \quad (3.7)$$

- Linear

$$y = a + b * ii \quad (3.8)$$

- Cubic Polynomial

$$y = b + c * TC + d * TC^2 + e * TC^3 \quad (3.9)$$

- Sinusoidal

$$y = a - b * \sin(ii * c + d) \quad (3.10)$$

- Gaussian

$$y = a + b * \exp(-0.5*(ii/c)^2) \quad (3.11)$$

where  $TC = ii/a$ ; and  $ii$  goes from 1 to the total number of values contained in the data.

### 3.2.2 Input variables

***NP*** (*number of population members*)

In statistics, the term population is used to describe the subjects of a particular study. The Differential Evolution (DE) method is a population based search technique where the number of variables used as population is NP. The initial population is randomly chosen if there is no prior information available about the problem [26].

If this parameter is increased, then the population is increased, so that means that the time it takes to perform the optimization is longer. But it also should not be too small, as that would affect to the final result, making it less accurate.

***D*** (*number of parameters of the objective function*)

D refers to the number of parameters of the objective function. As stated before, depending on the cost function used, the number of parameters needed to fulfill the equation is different. That number can be set using this variable D. It can go from 2 (needing only  $a$  and  $b$ ) to 5 (using all the parameters:  $a, b, c, d$  and  $e$ ).

***F*** *DE differential weight* ( $F \in (0, \infty)$ )

F is a weighting or scaling factor applied to the difference vectors to build the weighted difference vector. It controls the amplification of differential variations, determining the mutation step size: a lower F will result in smaller step sizes and the optimization will take longer to converge, whereas a higher value of F would facilitate the exploration but having the risk of overshooting the optimal solution [25].

To improve the convergence behavior, especially in noisy objective functions, it has been found that the optimal range of this factor is in the range of (0.5, 1.0) [23].

***CR*** *crossover probability constant* ( $CR \in (0, 1)$ )

The crossover constant CR represents the probability that the child vector inherits the parameter values from the noisy random vector [26]. It controls the number of elements of the parent vector ( $x_i(t)$ ) that will change. A higher value



of the crossover probability means that more variation is being introduced in the new population resulting in an increase of diversity and exploration. Normally, increasing the value of CR results in a faster convergence while decreasing it increases the search robustness [25].

***itermax*** *maximum number of iterations (generations)*

As it can be deduced by the variable name, **itermax** refers to the maximum number of generations or iterations that the function is allowed to run. If this number is reached, the function will come to an end.

With this parameters happens the same as in the case of NP, a higher value for **itermax** will mean that it takes more time, but it does not mean that it will return a better result. But, also, a low value of **itermax** may end up with a bad result as the function did not have enough number of generations to reach a good solution.

***strategy***

The type of **strategy** used to achieve the goal can be set to different parameters. The strategies vary according to three different parameters, so the general convention used to refer to strategy is of the type "*DE/x/y/z*" [26].

*DE* stands for Differential Evolution.

*x* is a string that represents the vector that will be perturbed, the perturbation can be in the best vector of the previous generation or in a randomly selected one [26].

*y* is the number of difference vectors considered for the perturbation of *x*, it can use a single vector or two difference vectors. In the case of using only one difference vector, out of three distinct randomly chosen vectors, the weighted vector differential of any two vectors is added to the third one, three random vectors are selected in the mutation operation, and the weighted vector difference of two of them is added to the third one; and in the case of using 2 difference vectors, five random vectors are selected instead of three and the weighted vector difference of each pair of any four vectors is added to the fifth one for perturbation [26]. By increasing the number of differential vectors used, the power of exploration of DE is being increased [25].

$z$  tells the type of crossover that is used which can be exponential or binomial [26]. It can be seen that in the strategies used in the program,  $z$  is not specified, this is because it is already defined within the program that it will perform a binomial crossover.

The different strategies used in the program could be chosen between the following ones:

`strategy = 1` → DE/best/1

`strategy = 2` → DE/rand/1

`strategy = 3` → DE/rand-to-best/1

`strategy = 4` → DE/best/2

`else` → DE/rand/2

Depending on the type of strategy chosen for the modeling, the differential variation is performed in a different way.

### ***data***

This last input variable refers to the data set the estimation is going to be performed on. In this case, this data represents the filtered impedance signal between the specific time points (Figure 3.4).

### **3.2.3 Output variables**

The function returns two output variables, which are *bestmem* and *nfeval*.

***bestmem*** *Parameter vector with best solution*

It is a vector of length  $D$  (number of parameters) that has stored in each position the best parameter found. This is the variable used to be able to construct the models following the corresponding equation of the ones mentioned above (depending on the type of estimation performed).

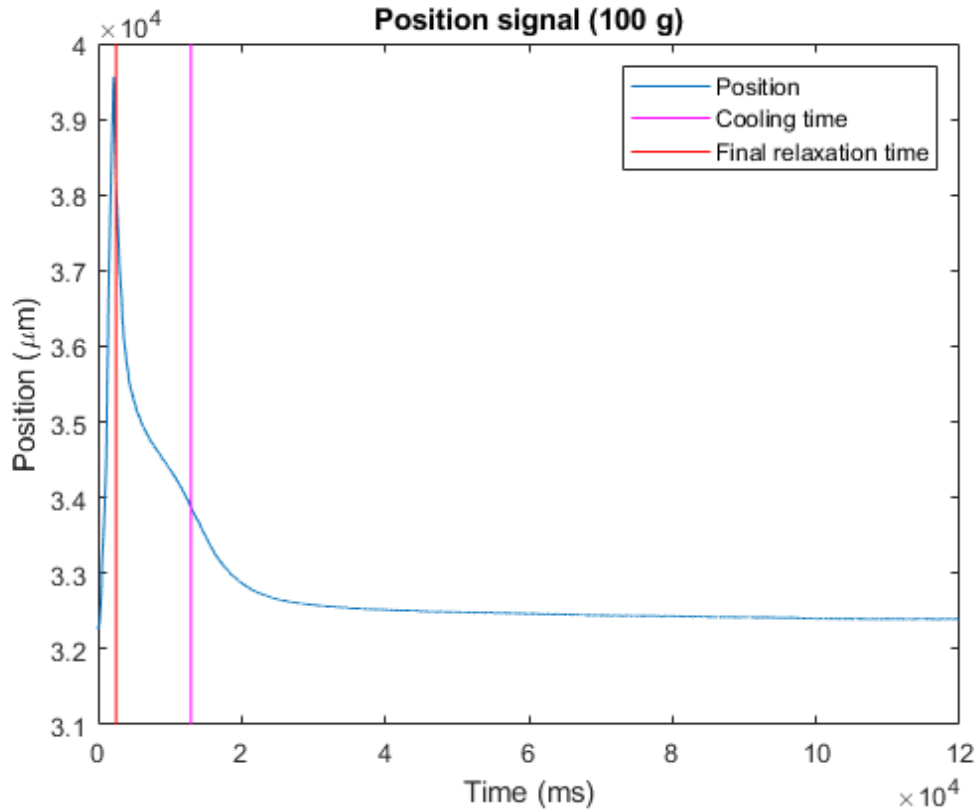
***nfeval*** *Number of function evaluations*

The implemented function also returns the number of function evaluations *nfeval*. This variable represents the number of times the variables were compared to the cost function to check which one should be included in the following iteration.

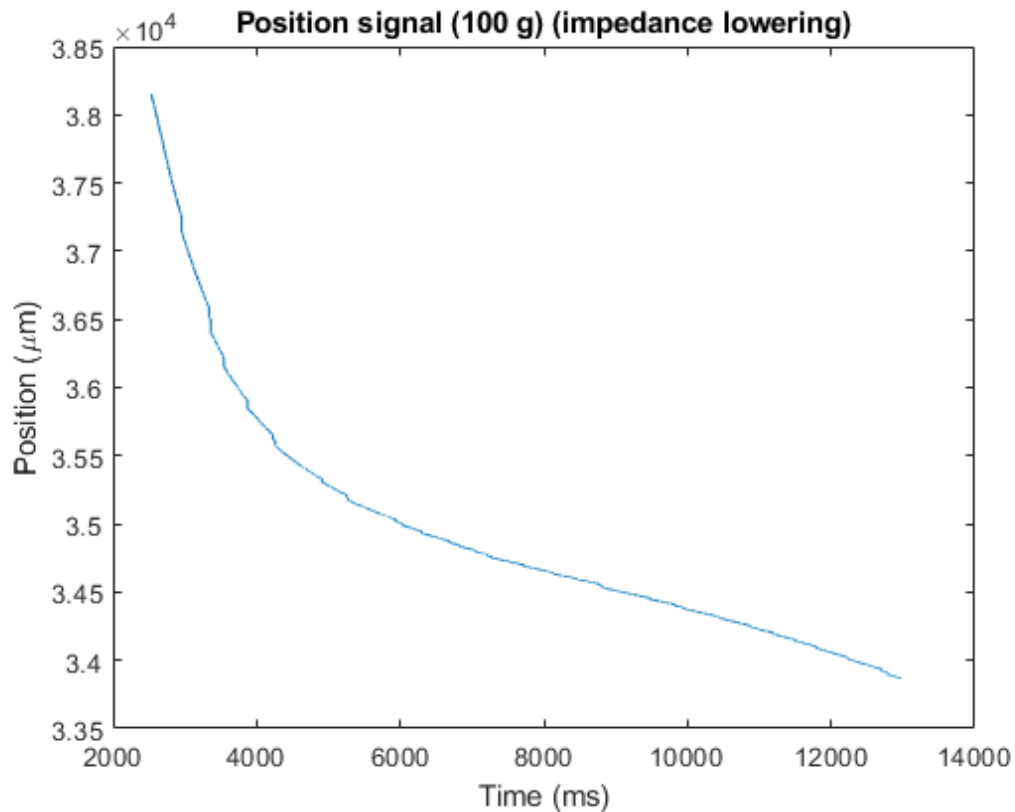
### 3.2.4 Screen output

Besides the two output variables returned by the function, the optimization function also prints in the screen, every 100 iterations, the best cost value (**Best**) up to that time as well as the vector containing the best solution. Once the program finishes, it also prints the elapsed time (this value will also be increased a little by the printing of the variables on the screen). The values of the scaling factor **F**, the crossover probability **CR** and the number of population members **NP** are also printed, but they do not provide any new information as in the case of this function they are kept constant.

For the position there is no need to either apply the optimization to build a model or to filter the signal, as the data itself is very clean as it can be seen in Figure 3.7, where the position of the SMA bearing a load of 100 g is shown. The position signal focused in the analysis time can be seen following, in Figure 3.8



**Figure 3.7:** Position signal of a load of 100 g. The red and magenta vertical lines represent the times the modeling is focused in (lowering analysis between cooling time (red) and final relaxation time (magenta)).



**Figure 3.8:** Position signal of 100 g of load between cooling time and final relaxation time.

### 3.3 Testing the models

The models are tested at different time points to check the accuracy at each one using the other impedance signal that was left for this purpose from the samples that were given.

The value of the impedance is needed at each of the time points, but it can not be done with just one value because as it was explained in the introduction, the material has hysteresis, there are at least two points with the same position value or hysteresis as the material is relaxed or contracted. So, a sequence of impedance values was taken over a time interval (10 values over time intervals of 25 ms and 50 ms) to be able to calculate its load and its position.

### 3.3.1 Load

To perform the tests for the 17 loads, another impedance signal is needed for each of them. To get better results, this signal had to be filtered prior to the taking of the 10 impedance values, so that again the outliers are not taken into account because they may lead to a wrong estimation.

After the impedance values are taken in their time range, they are compared with the values of the models that are stored (comparing the same time period). Each real data of impedance will be compared to the 17 models that were created. The method selected to measure the similarity between real data and the models is the sum of squared differences (SSD).

$$SSD = \sum (p_i - q_i)^2, \quad (3.12)$$

where  $p_i$  represents impedance values taken from the test data and  $q_i$  represents impedance values from the models

The sum of squared differences takes into account all the values selected in the time range of both sets as it can be seen in equation 3.12. In fact, it is calculating the square of the distance between each point and then summing all the values. So, the lower the value returned by the SSD will mean that the distance between the two sets is lower, so they will be the most similar. So, in order to find the load the data belongs to, the minimum value has to be selected among the 17 values that are returned. As it has been previously stated, the range of loads the signal may belong to, is being estimated instead of the precise load, so in the end, the two lowest values of the SSD are taken into account.

### 3.3.2 Position

There are two options to estimate the position:

- The first option will take into account the previously estimated load so that then one can easily go to the position model that corresponds to that load. It is likely for this estimation to be dependent on the accuracy of the load estimation, so, it is expected that this will work well if the estimation of the

load presents a high accuracy, otherwise the position estimation will also be inaccurate. That is why another option was tested.

- This second method is used to see if knowing the load the signal belongs to, one can find out its position only by looking at the impedance values, not knowing the specific time point, but knowing the phase in which the material is in: if it is in the heating phase or in the cooling one. As it has been mentioned, in the project the analysis is focused in the cooling phase, so this position estimation will also be performed on that one.

In this case, the sequence of 10 impedance values will be compared to the model created for that specific charge in order to find the correct time point so that then the position can be estimated. It will be tested at the same 4 different time points as the previous tests for all the loads.

In both cases, the position returned will correspond to that of the time points that have been selected ( $t = 4408$  ms,  $t = 6908$  ms,  $t = 9408$  ms,  $t = 11908$  ms).

## 4 Results & Discussion

Under this section, the results obtained by applying the previously explained methods will be discussed with the aid of some figures and tables.

### 4.1 Modeling

In this section, the results of the modeling will be discussed showing the different cost functions that were used to test the optimal one. The following results shown in Figure 4.2 corresponding to the modeling belong to a load of 100 g and they were built using the final parameters chosen, which are the following:

NP	20
D	5
F	0.8
CR	0.7
itermax	5000
strategy	2

**Table 4.1:** Input parameters of the estimation function.

These parameters were chosen on basis of the accuracy of the model returned and the time needed to perform the optimization. For example, if a higher NP would have been used, it would have returned the same model but taking more time; and the same would have happen increasing `itermax`. The number of parameters used for the objective function (D), is set to 5 so that all the cost functions can be tried at the same time without having to change this parameter, but after the decision to use the exponential cost function is made, D can be changed to 4, as it is the number of parameters needed to fulfill equation 3.4. In the case of the differential evolution stepsize (F) and the crossover probability (CR), big modifications of these values led to different shapes in the modeling that did not correspond with the original data. F was chosen in the range that was specified as the optimal range: [0.5, 1].



As it was said, there are also some variables printed in the screen, an example of this is shown in Figure 4.1, where it can be seen that it takes approximately 2 seconds to perform each optimization. As mentioned, also CR, F and NP are also printed but they are constant throughout every iteration.

```
Iteration: 4900, Best: 0.433028, F: 0.800000, CR: 0.700000, NP: 20
best(1) = 5.998465
best(2) = -340.683348
best(3) = 88.344275
best(4) = 7.117937
best(5) = -282.134157
Elapsed time is 1.907812 seconds.
```

**Figure 4.1:** Variables printed on the screen during optimization.

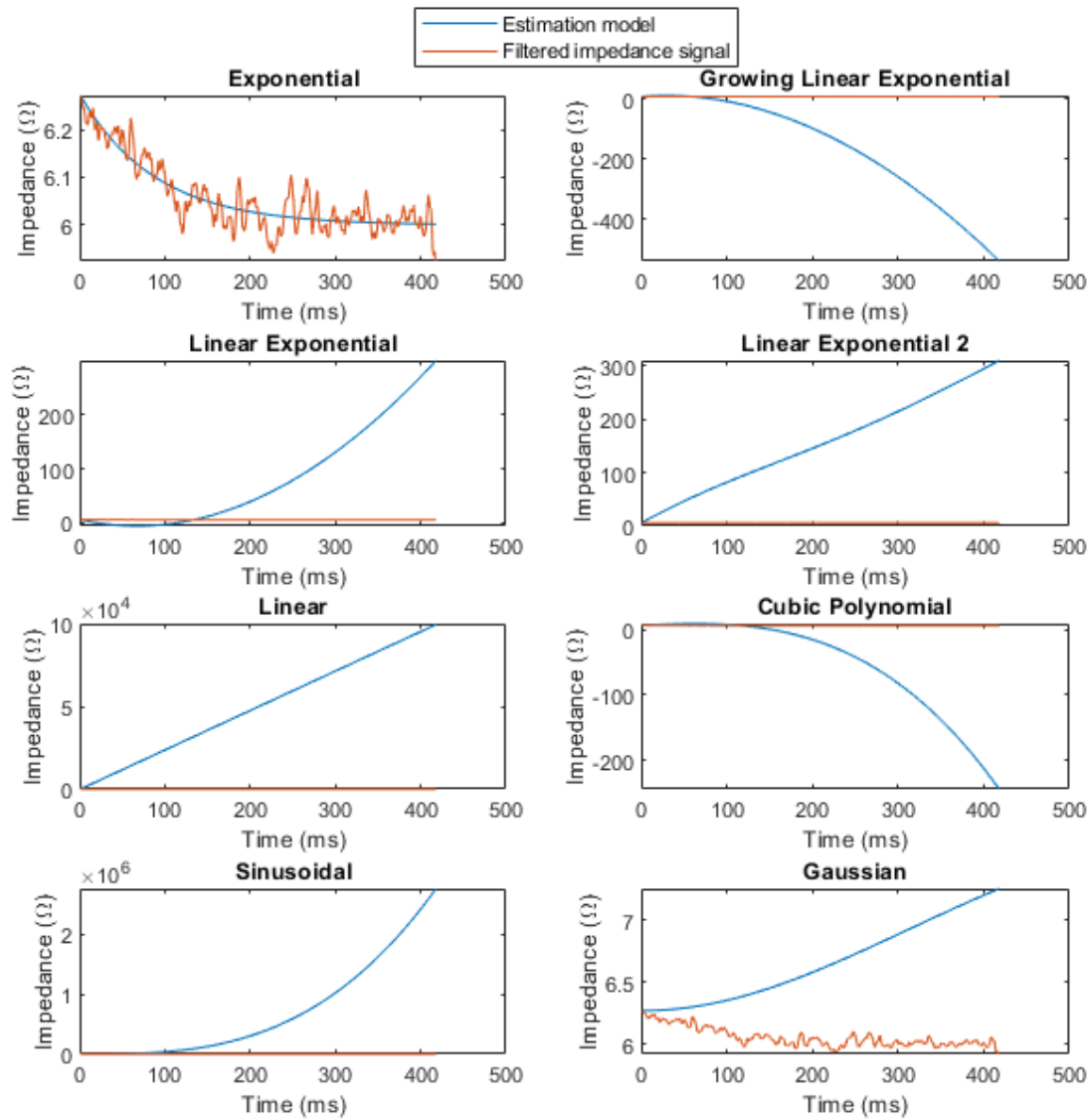


Figure 4.2: Estimation models.

As it can be observed in Figure 4.2, the only estimation model that belongs to the same range and follows the trajectory of the real data is the one that uses the exponential cost function (first graph in the left column). The other cost functions cover a wider range in the y axis (Impedance), which does not correspond to the one from real data. So, the modeling of the impedance response to be used in the estimation of load and position will be the one corresponding with the exponential cost function (Equation 3.4) and the values of the variables shown in Table 4.1.

In the following figure, Figure 4.3 the 17 models that were built are shown

alongside with the filtered data in the range between cooling time and final relaxation time to show that in all cases the modeling was performed correctly following the trajectory of the input data. These models are performed using the exponential estimation cost function because as it was seen in the previous figure, it was the one that yielded the best results.

As seen in the figure, the models follow the trajectory of the filtered data. A measure of the SSD was also performed between the models and the signal to quantify this similarity and it is shown on Table 4.2. The results obtained are what is expected, not high but still not really close to 0 which is what would have been optimum. This is expected because the filtered signal has still some fluctuations that could not be erased without messing the signal, so it is an error that will have to be kept in mind when doing the estimations.

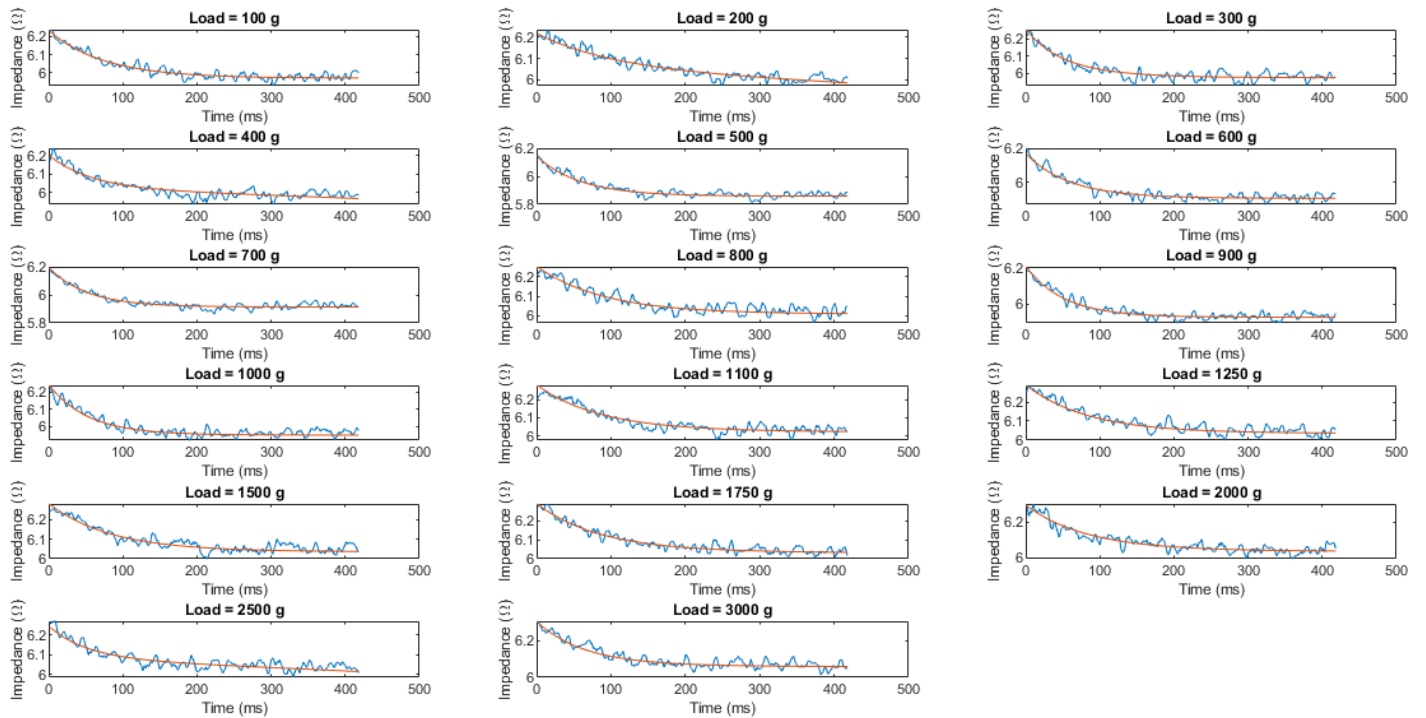


Figure 4.3: Estimation models for all loads.

Load (g)	SSD (g)
100	0.14175
200	0.24131
300	0.19834
400	0.20491
500	0.19445
600	0.12495
700	0.21919
800	0.20459
900	0.19404
1000	0.14165
1100	0.23062
1250	0.21169
1500	0.16708
1750	0.16584
2000	0.14273
2500	0.1334
3000	0.19389

**Table 4.2:** Sum of Squared Differences between model and signal.

## 4.2 Load estimation

The models that were created, are tested with some data to see what is the range of error when estimating the load it is bearing and its position. So, there are two estimations, first the load is assigned and afterwards the position.

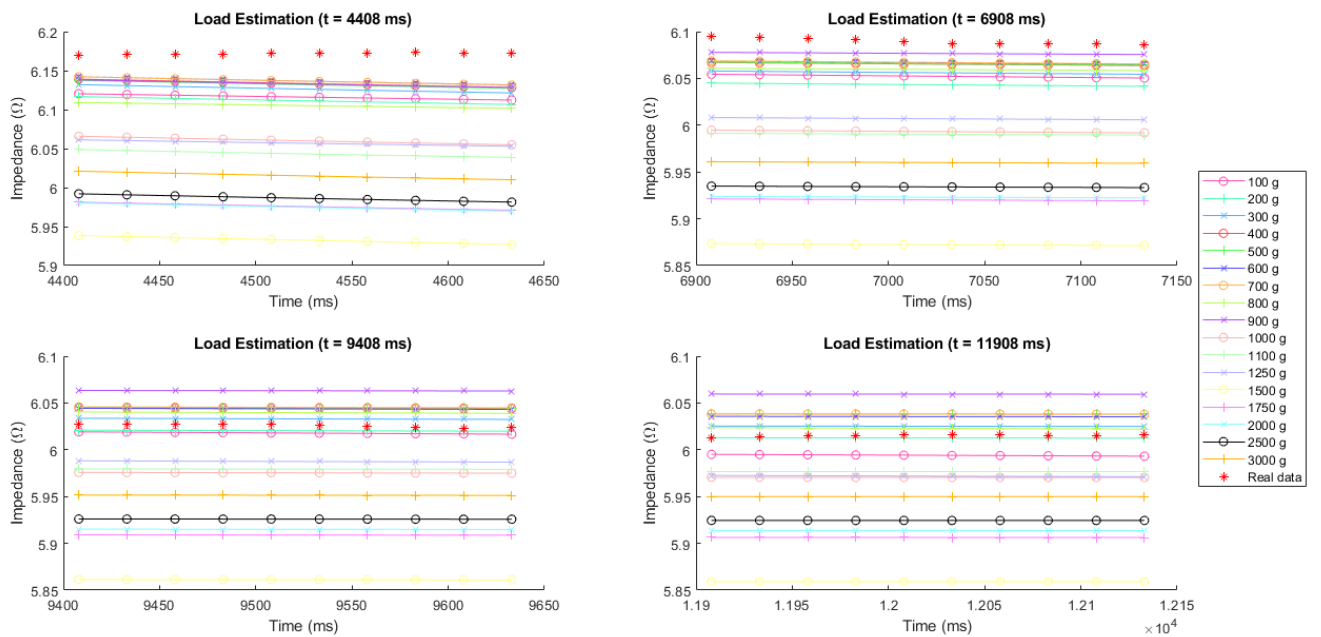
To do so, a sequence of impedance values are collected in a time interval, and they are correlated with the models that were previously built for the different loads by calculating the sum of squared differences between them and selecting the one that returns a minimum value. That model will be the most similar to the data that was tested. In order to get a range of load, as it is very possible that the data tested does not correspond exactly to one of the models, a second minimum is taken.

The values of the impedance were taken at 4 different time points ( $t = 4408$  ms,  $t = 6908$  ms,  $t = 9408$  ms,  $t = 11908$  ms) to test the accuracy of the system

over time. At each time point, 10 values of impedance were taken in intervals of 25 ms (for each of the loads) and they were compared to all 17 models that were previously obtained. The results of the load assignment for each case are shown in Table 4.3

The project was tested by taking different numbers of impedance values (ranging from 5 to 20), but taking a number of impedance values lower than 10 yielded for a worst estimation of load and taking a number of impedance values higher than 10 returned approximately the same result but taking more time, so the best solution was achieved by taking a sequence of 10 impedance values.

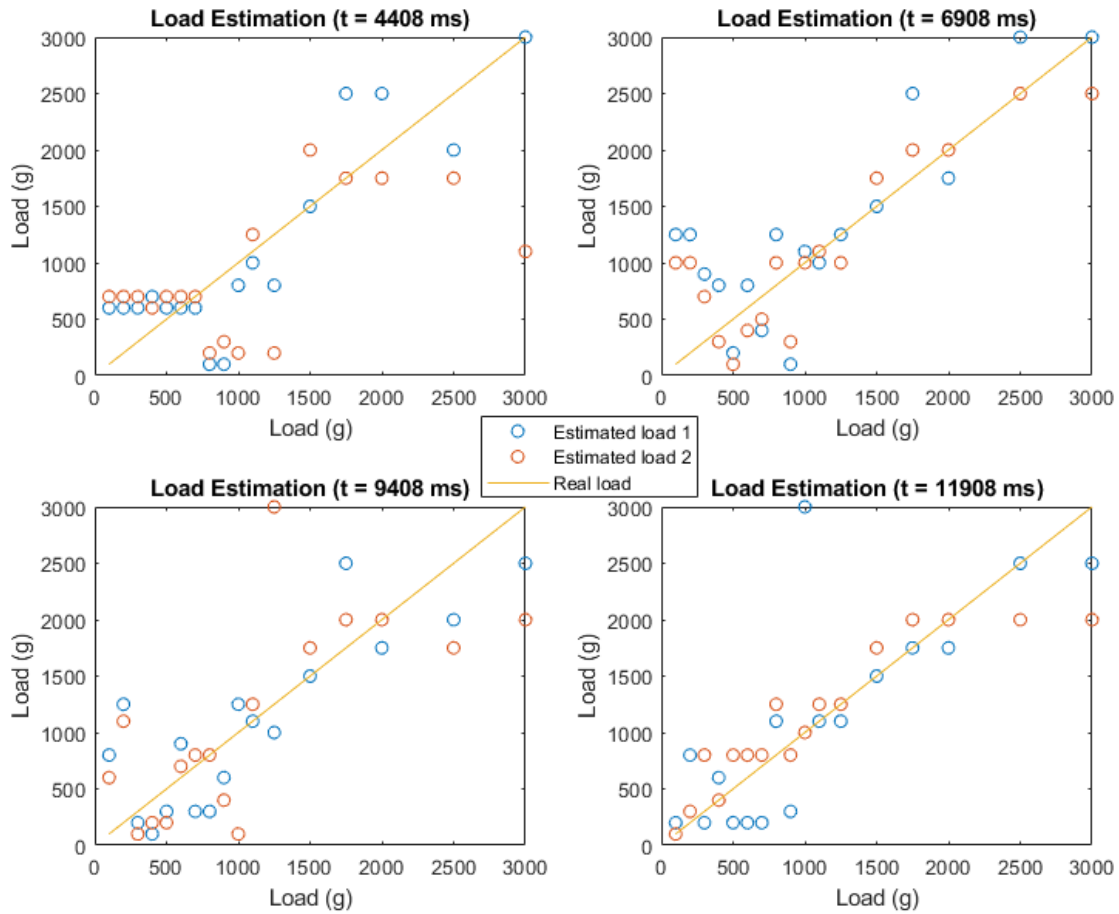
An example of the signals being compared is shown in Figure 4.4, where it can be seen for the 4 different times, the 10 values of impedance that were taken from the real data: the ones that are being evaluated; and the 10 values corresponding to the 17 different models of impedance that were created. The load value of the real data shown corresponds to 100 g.



**Figure 4.4:** Impedances for load estimation at the 4 different times stated.

Real load (g)	Estimated load (g) ( $t = 4408$ ms)		Estimated load (g) ( $t = 6908$ ms)		Estimated load (g) ( $t = 9408$ ms)		Estimated load (g) ( $t = 11908$ ms)	
100	600	700	1250	1000	100	200	800	300
200	600	700	100	200	1250	1100	300	800
300	100	200	900	700	200	100	900	500
400	900	400	200	100	1250	100	800	300
500	300	100	200	1250	100	200	800	300
600	600	700	900	700	900	700	100	1100
700	900	500	200	100	900	700	200	100
800	200	100	1250	200	1250	100	100	1100
900	300	500	900	700	900	700	900	500
1000	800	200	1000	1100	100	200	3000	2500
1100	1100	1250	2500	2000	100	200	200	100
1250	200	800	1100	1000	2500	2000	100	200
1500	1500	2000	1500	1750	1500	1750	1500	1750
1750	2500	3000	1750	2000	3000	2500	1750	2000
2000	2500	1750	1500	1750	1750	2000	1750	2000
2500	1500	2000	3000	1100	2500	2000	2500	2000
3000	1100	1250	3000	2500	1750	2000	3000	1000

**Table 4.3:** Load estimation for the 4 different times stated. Highlighted in blue are the real positions that belong to the range of the estimated position.

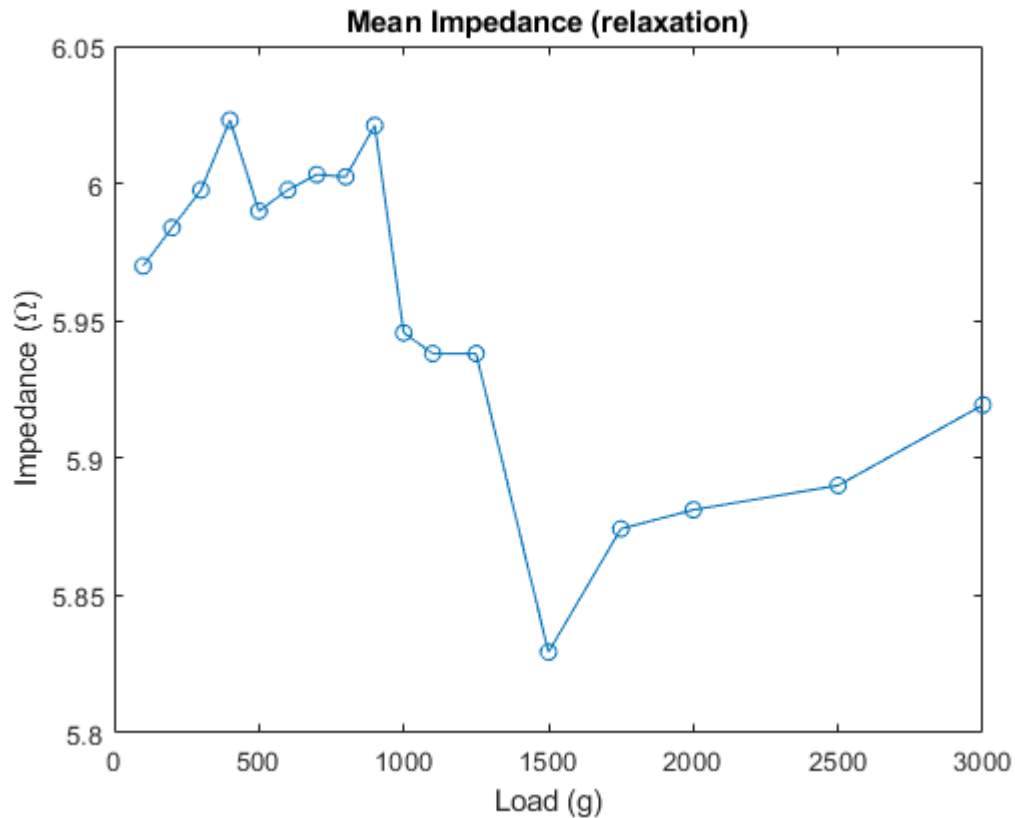


**Figure 4.5:** Load estimation at 4 different times showing the real load (in yellow) and the two estimated loads (in blue and red).

In Table 4.3, it can be seen that as time increases, the estimation of the load becomes more accurate, being in the relaxation state where the most accurate results are obtained. A graph containing the estimations showed in the previous Tables is found in Figure 4.5, where the improvement of the estimation along time is more noticeable, seeing the estimated loads almost form a cluster along the line they should be following (real load) in the three last time points ( $t = 6908$  ms and  $t = 9408$  ms and  $t = 11908$  ms), whereas for the lowest time ( $t = 4408$  ms), the estimated loads are almost the same for all the different real loads, especially for smaller loads, beginning to make a difference at time  $t = 6908$  ms where the estimation for big loads start to approximate to the real one and the small loads start to become more separated taking different values. This could be due to the state of transition the material is in (it is changing from austenite phase to martensite) or because in the modeling step, the initial part (corresponding

to lower times) could have been less accurate as it was still trying to find the appropriate solution by using a random initialization, but then it stabilizes as time passes. So, maybe, using an initialization closer to the expected result could improve the modeling in the beginning which may lead to a better estimation of the load at those times.

In this relaxed state, the differences between the loads are more significant, which means that they can be more easily distinguishable (at least in a roughly manner). A separation between big loads and small loads could be made when the SMA is relaxed. This can be checked plotting the mean of the original impedance signal for each of the loads in this period of time (relaxation). In Figure 4.6, it can be observed that the higher the load, the impedance decreases, so a distinction between small and big loads could be obtained.



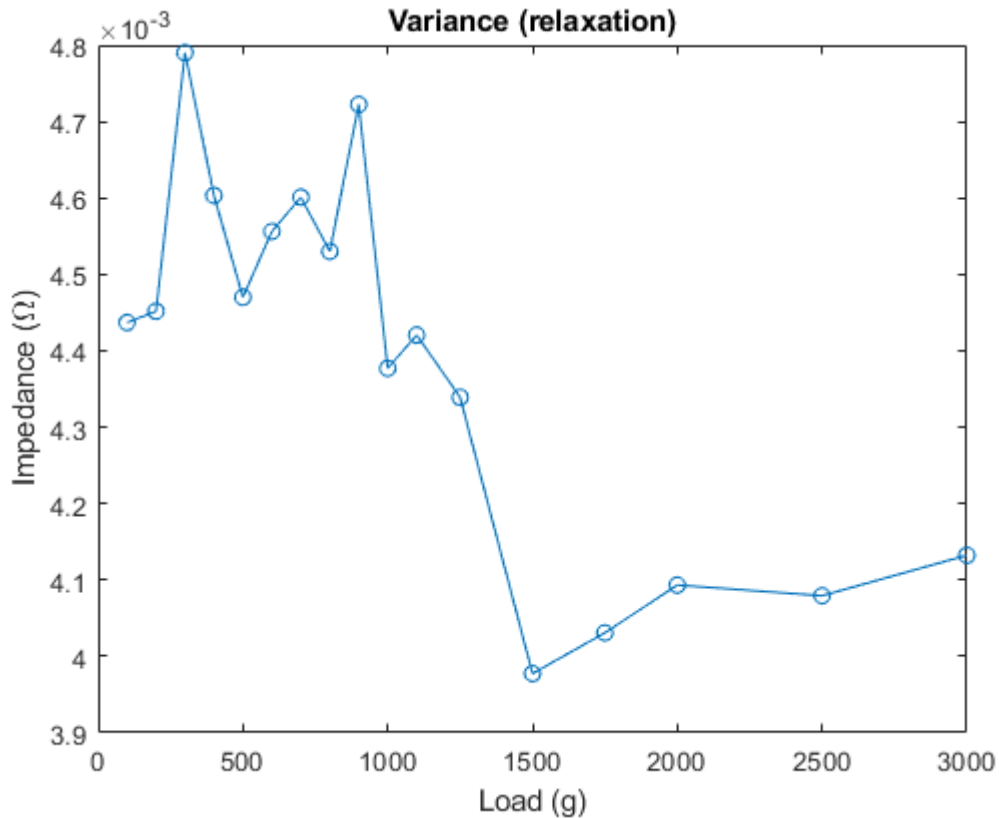
**Figure 4.6:** Mean impedance during relaxation of the different loads studied.

Some other options that were tried to obtain a difference between loads were to measure the variance in the relaxed part of the signal to see if there were any significant differences between low and high loads meaning that this variable was



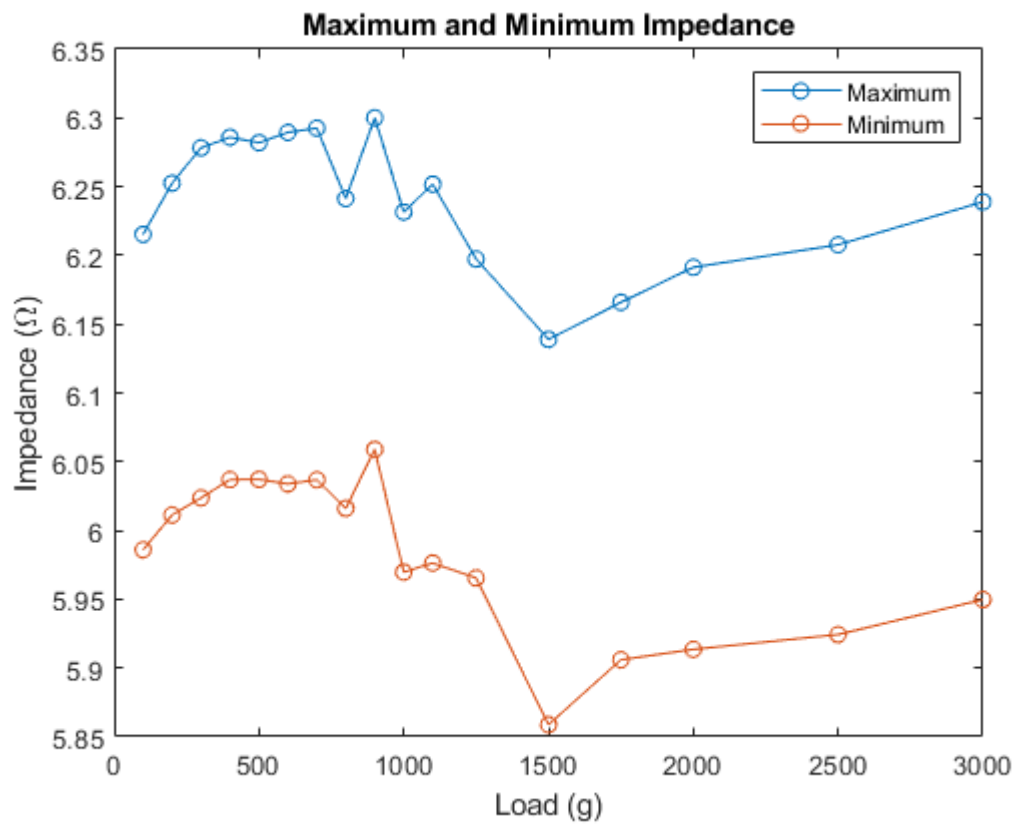
dependent on them; and the measurement of the maximum and minimum of the signals to see if the difference between loads was significant between them.

The results showed otherwise, as it can be seen in Figure 4.7, the variance of the different loads vary in a really small range ( $10^{-3}$ ), so the fluctuations in the signal can be attributed to the set up of the experiment and they are not seen as something characteristic of the amount of load supported by the SMA wire.

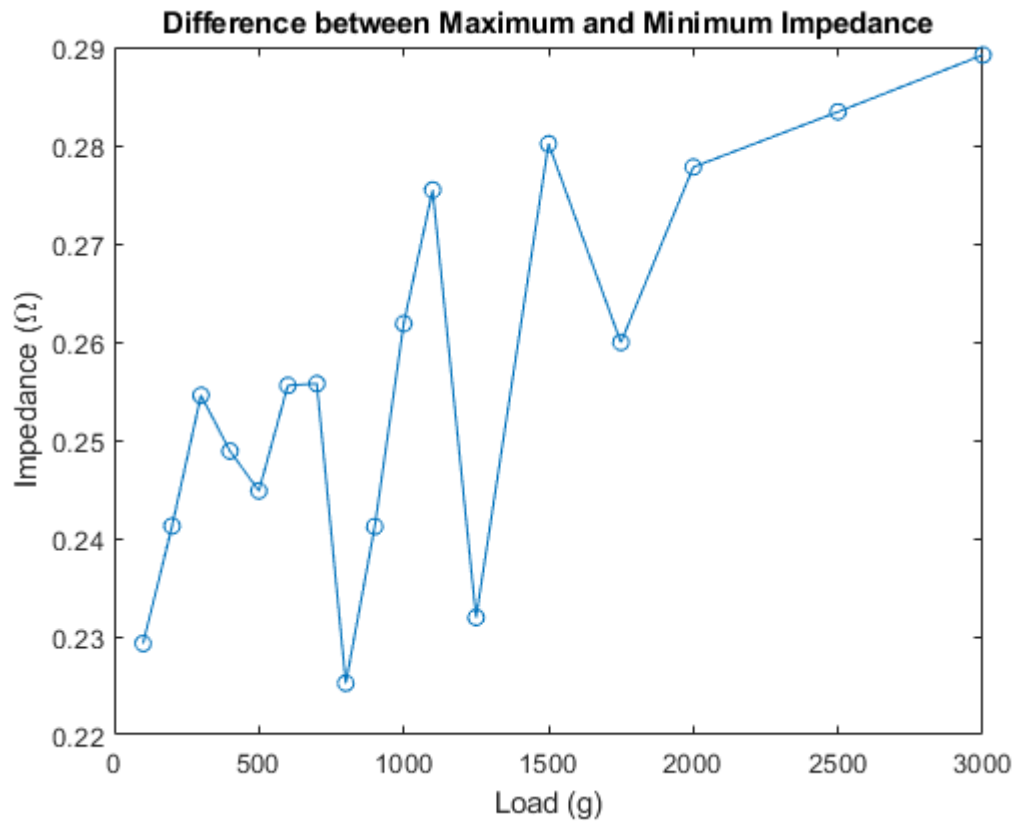


**Figure 4.7:** Variance of the impedance during relaxation of the different loads studied.

Similarly, in Figure 4.8, it is shown the graph for the maximum and minimum of the 17 different loads tried, and it can be observed that the differences between maximum and minimum for each of them are more or less constant as it shown in the following plot (Figure 4.9), with values going from 0.225 up to 0.29 which is not a range that can be considered significant.



**Figure 4.8:** Maximum and minimum values of the impedance of the different loads studied.



**Figure 4.9:** Difference between maximum and minimum values of impedance of the different loads.

### 4.3 Position estimation

Two different tests are performed to estimate the position of the SMA muscle. The first one is a follow up of the previous test, the load estimation, where the position is estimated based on the assignment of the load. And the second one is independent of the estimated load, it uses the real load to find out the estimated position.

#### 4.3.1 Dependent on load estimation

The position of the SMA muscle is tested after the estimation of load has been performed. Once a load is assigned to the signal the position can be selected from

the position data corresponding to that load by using an impedance value at a specific time point. In this case there will also be a range of estimated positions, because the position will be selected from the previously estimated load, so as in the estimation of the load a range was selected, now for each of those loads the position will be estimated.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )		Range width ( $\mu\text{m}$ )
100	35424.9	34453.17	34257.6	195.57
200	3307.18	34453.17	34257.6	195.57
300	3125.15	35424.9	3297.05	32127.85
400	19172.3	34087.78	19162.79	14924.99
500	34648	3125.15	35414.77	32289.62
600	34453.17	34453.17	34257.6	195.57
700	34265.9	34087.78	34640.44	552.66
800	34177.2	3307.18	35414.77	32107.59
900	34087.78	3125.15	34640.44	31515.29
1000	34134.99	34177.2	3297.05	30880.15
1100	33510.11	33510.11	33212.67	297.44
1250	33216.82	3307.18	34169.15	30861.97
1500	34403.15	34403.15	33347.6	1055.55
1750	33936.38	32355.62	31470.51	885.11
2000	33350.89	32355.62	33926.98	1571.36
2500	32355.62	34403.15	33347.6	1055.55
3000	31475.15	33510.11	33212.67	297.44

**Table 4.4:** Position estimation at time  $t = 4408$  ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )		Range width ( $\mu\text{m}$ )
100	34748.29	32753.83	33487.41	733.58
200	2634.96	34748.29	2631.42	32116.87
300	2449.27	33426.9	33633.33	206.43
400	18528.38	2634.96	34745.97	32111.01
500	34025.55	2634.96	32748.46	30113.5
600	33833.16	33426.9	33633.33	206.43
700	33637.84	2634.96	34745.97	32111.01
800	33549.15	32753.83	2631.42	30122.41
900	33426.9	33426.9	33633.33	206.43
1000	33492.05	33492.05	32915.36	576.69
1100	32920.48	31824.43	32906.33	1081.9
1250	32753.83	32920.48	33487.41	566.93
1500	34038	34038	33521.33	516.67
1750	33525.84	33525.84	32906.33	619.51
2000	32909.26	34038	33521.33	516.67
2500	31824.43	30917.12	32915.36	1998.24
3000	30917.12	30917.12	31818.94	901.82

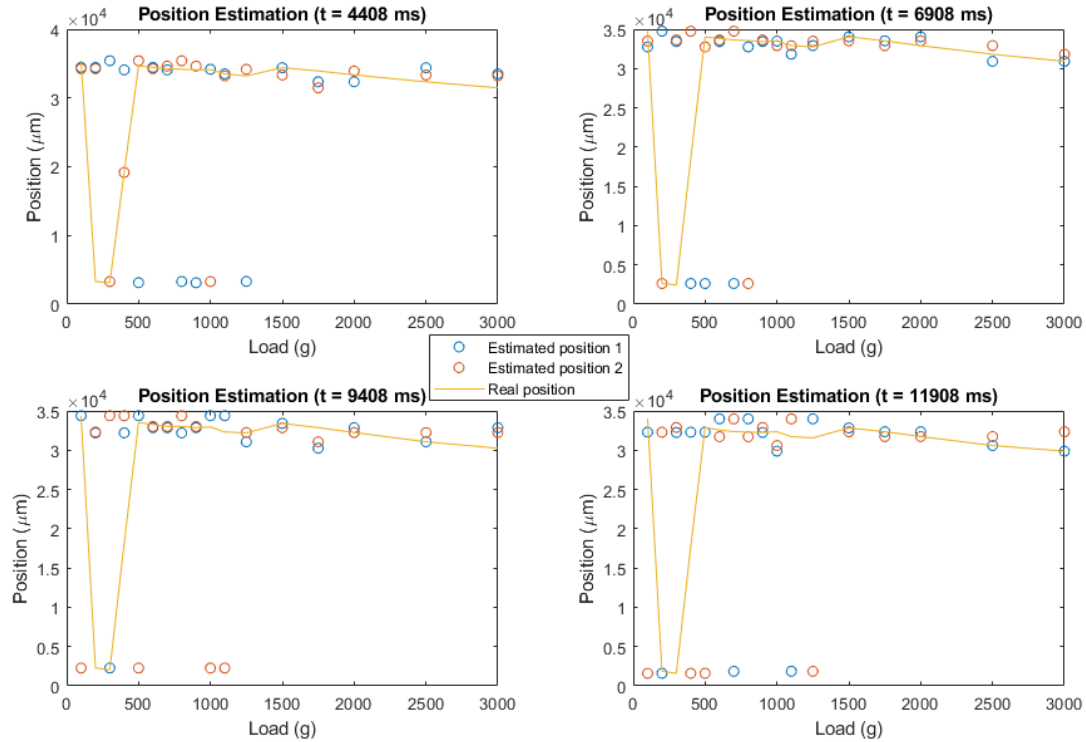
**Table 4.5:** Position estimation at time  $t = 6908$  ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )		Range width ( $\mu\text{m}$ )
100	34392.16	34392.16	2269.44	32122.72
200	2272.86	32204.46	32307.06	102.6
300	2062.04	2272.86	34389.24	32116.38
400	18134.69	32204.46	34389.24	2184.78
500	33522.43	34392.16	2269.44	32122.72
600	33274.41	32865.21	33011.98	146.77
700	33016.25	32865.21	33011.98	146.77
800	33036.14	32204.46	34389.24	2184.78
900	32865.21	32865.21	33011.98	146.77
1000	32956.71	34392.16	2269.44	32122.72
1100	32321.95	34392.16	2269.44	32122.72
1250	32204.46	31067.42	32245.46	1178.04
1500	33410.43	33410.43	32860.21	550.22
1750	32866.68	30250.14	31063.27	813.13
2000	32251.07	32866.68	32245.46	621.22
2500	31067.42	31067.42	32245.46	1178.04
3000	30250.14	32866.68	32245.46	621.22

**Table 4.6:** Position estimation at time  $t = 9408$  ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )		Range width ( $\mu\text{m}$ )
100	33990.42	32293.77	1597.47	30696.3
200	1854.77	1601.49	32283.52	30682.03
300	1601.49	32256.19	32898.89	642.7
400	17683.05	32293.77	1597.47	30696.3
500	32903.52	32293.77	1597.47	30696.3
600	32565.1	33990.42	31722.07	2268.35
700	32370.75	1854.77	33986.88	32132.11
800	32293.77	33990.42	31722.07	2268.35
900	32256.19	32256.19	32898.89	642.7
1000	32350.74	29876.58	30592.72	716.14
1100	31727.32	1854.77	33986.88	32132.11
1250	31567.13	33990.42	1850.74	32139.68
1500	32831.66	32831.66	32346.71	484.95
1750	32350.38	32350.38	31735.86	614.52
2000	31740.5	32350.38	31735.86	614.52
2500	30593.94	30593.94	31735.86	1141.92
3000	29876.58	29876.58	32346.35	2469.77

**Table 4.7:** Position estimation at time  $t = 11908$  ms. Highlighted in blue are the real positions that belong to the range of the estimated position. The last column shows the range width of the estimated positions.



**Figure 4.10:** Position estimation at the 4 different times stated.

Contrary to what was expected, the results of the position estimation seen in Tables 4.4 to 4.7 show a better estimation than the ones coming from the load, at least by counting how many times the real load or position belonged to the estimated range. But, taking a closer look into the results, it can be seen that in some cases this estimated range is really wide (range width higher than  $3000 \mu\text{m}$ ).

Also, it can be observed that this estimation does not really improve or deteriorate over time, but it can be noted that at time  $t = 4408 \text{ ms}$ , the estimated position does not correspond to the real one, which makes sense as at that time point the load estimation for small loads was not accurate.

### 4.3.2 Non dependent on load estimation

As it has been said, this method does not rely on how good or bad the estimation of the load was. It is basically done to check the relation between the impedance and the position of the SMA muscle. The following tables (Tables 4.8 to 4.11) and figure (Figure 4.11) show the results obtained when performing the estimation fol-



lowing this second method. Supplementing the tables with the figure allows for a better vision of the similarity of the results obtained. In this case, there are no highlighted values in the tables as the estimated position does not correspond to a range of values, that is the reason why the SSD is added, to show a quantitative measurement of the similarity between the estimated values and the real ones.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )	SSD
100	35424.9	36638.92	0.00023
200	3307.18	4017.83	0.00037
300	3125.15	3669.76	0.00032
400	19172.3	19529.27	0.00029
500	34648	35173.21	0.00033
600	34453.17	35029.49	0.00042
700	34265.9	34855.04	0.00036
800	34177.2	35099.4	0.00041
900	34087.78	34768.78	0.00024
1000	34134.99	34953.86	0.00038
1100	33510.11	33850.85	0.00071
1250	33216.82	33619.18	0.00067
1500	34403.15	34569.07	0.00046
1750	33936.38	34105.59	0.00064
2000	33350.89	33531.09	0.00082
2500	32355.62	32508.36	0.00063
3000	31475.15	31640.7	0.00088

**Table 4.8:** Position estimation at time  $t = 4408$  ms. Also shown the value of SSD in the third column.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )	SSD
100	34748.29	35131.24	2.00E-05
200	2634.96	3097.21	2.00E-05
300	2449.27	2997.78	1.00E-05
400	18528.38	18791.66	3.00E-05
500	34025.55	34448.77	4.00E-05
600	33833.16	34279.93	1.00E-05
700	33637.84	34288.59	1.00E-05
800	33549.15	33997.98	2.00E-05
900	33426.9	33732.88	0.00E+00
1000	33492.05	33990.67	1.00E-05
1100	32920.48	33501.08	1.00E-05
1250	32753.83	33221.21	0.00E+00
1500	34038	34295.05	1.00E-05
1750	33525.84	33887.82	1.00E-05
2000	32909.26	33300.88	6.00E-05
2500	31824.43	32240.94	0.00E+00
3000	30917.12	31316.91	3.00E-05

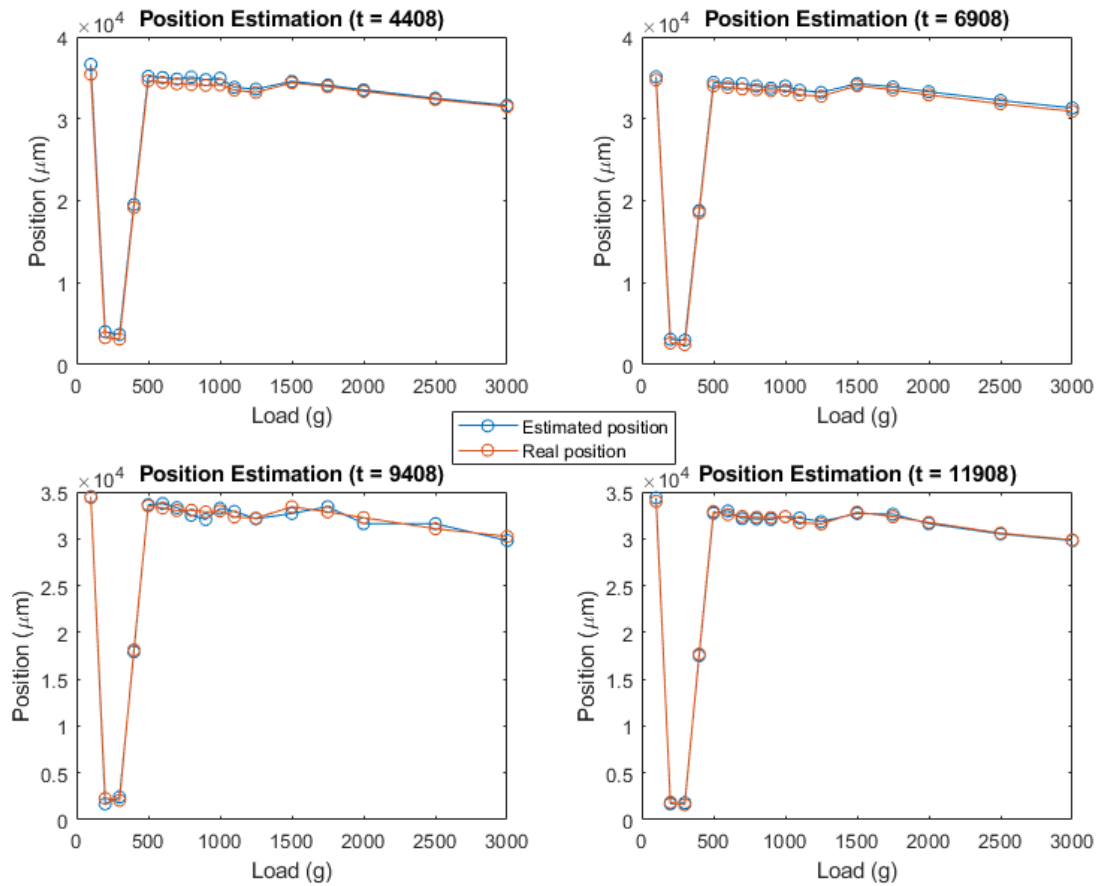
**Table 4.9:** Position estimation at time  $t = 6908$  ms. Also shown the value of SSD in the third column.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )	SSD
100	34392.16	34491.11	1.00E-05
200	2272.86	1688.72	3.00E-05
300	2062.04	2427.8	1.00E-05
400	18134.69	17918.99	1.00E-05
500	33522.43	33630.64	3.00E-05
600	33274.41	33747.52	1.00E-05
700	33016.25	33315.76	3.00E-05
800	33036.14	32508.37	2.00E-05
900	32865.21	32049.04	2.00E-05
1000	32956.71	33245	1.00E-05
1100	32321.95	32910.11	2.00E-05
1250	32204.46	32133.46	3.00E-05
1500	33410.43	32696.24	0.0001
1750	32866.68	33421.04	1.00E-05
2000	32251.07	31611.18	7.00E-05
2500	31067.42	31603	0.00E+00
3000	30250.14	29802.4	0.00137

**Table 4.10:** Position estimation at time  $t = 9408$  ms. Also shown the value of SSD in the third column.

Real load (g)	Real position ( $\mu\text{m}$ )	Estimated position ( $\mu\text{m}$ )	SSD
100	33990.42	34355.32	3.00E-05
200	1854.77	33972.71836	0.00129
300	1601.49	1581.974023	2.00E-05
400	17683.05	32267.04824	0.00295
500	32903.52	1830.683215	0.00391
600	32565.1	32876.36523	1.00E-05
700	32370.75	1581.974023	0.00047
800	32293.77	1830.683215	0.00038
900	32256.19	32876.36523	0.00148
1000	32350.74	32323.40039	1.00E-05
1100	31727.32	31705.25039	0.00E+00
1250	31567.13	31705.25039	1.00E-05
1500	32831.66	32812.19336	0.00069
1750	32350.38	32336.4541	0.00E+00
2000	31740.5	32336.4541	0.00112
2500	30593.94	31717.24316	0.00036
3000	29876.58	31717.24316	0.00991

**Table 4.11:** Position estimation at time  $t = 11908$  ms. Also shown the value of SSD in the third column.



**Figure 4.11:** Position estimation at the 4 different times stated.

Looking at Figure 4.11, it can be observed that this second method yielded better results when estimating the position. This similarity was calculated by using the SSD, which can be seen in the last column of Tables 4.8 to 4.11. By looking at the results it is clearly noticeable how similar the real position and the estimated one are, as the SSD in all cases is really close to 0, meaning that they are almost identical.

## 5 Conclusions

Several conclusions can be extracted from the presented results keeping in mind the objective that was set at the beginning of the project. The objective was to successfully estimate both the load and the charge of a SMA muscle using the impedance responses, and this goal was broken down into three main steps: modeling, estimation of load and position estimation. So the conclusions can also be presented for each of these steps. In addition, several aspects that could be tried in the future are also stated.

### 5.1 Accuracy of modeling

The models obtained from the impedance responses followed the desired trajectory, the same as the impedance curve. The constraint of this method that is appreciated in this step of the project, as seen in Figure 4.3, is that the input signal is a really noisy one, so there will always be values above and below the model when performing the estimation that could lead to a wrong solution. But, in the end, using the exponential cost function, the models follow the trajectory that correspond to their input impedance signal.

### 5.2 Accuracy of load estimation

When estimating the load, a great difference was observed in the accuracy of the method between the times that were studied. For smaller times, closer to the cooling time (at the beginning of the transformation between austenite and martensite), more errors are observed than when this transformation has progressed until the SMA is in fully martensitic phase.

The load estimation for lower times should be improved. One solution that could be tried is to modify the initialization in the modeling, trying to initialize the population closer to the expected results so that the beginning of the modeling (lower times) could be improved, because the system tends to be more stable as

it starts to approach the right solution, so if at the beginning it is really far, the estimation will not be the best. But, actually, the main problem observed in the estimation was the noise coming from the impedance signal, so applying a different filter to the signal that would reduce the amount of outliers, the estimation could be improved.

### **5.3 Accuracy of position estimation**

The accuracy of the position estimation was dependent on the method used to perform the estimation, as it was expected.

The position signal varies with respect to the load, which means that having a wrong estimation of the load will probably cause a wrong estimation of the position. So, using the first method, the errors of the load estimation were accumulated to the errors that the own estimation of position may have, leading to considerable errors in the estimation.

In contrast, using the second method, knowing the precise load corresponding to the signal, the estimation is based only on the values of the impedance in an unknown time range, which is the variable that must be obtained in order to get the value of the position for that SMA wire at that time. So, removing the error coming from the load estimation, the accuracy of the position estimation is greatly increased.

### **5.4 Future work**

As it has been stated, the results obtained using this method were not completely satisfactory, so some improvements could be made to find the best way to estimate the position and the load of a muscle made of SMA. The improvements can be applied into the different steps where the signal was treated, which are the following:

### 5.4.1 Improving modeling and estimation

To improve either the modeling or the estimation, what can be changed is the filtering done on the signal or the type of estimation used.

- Filtering

For the filtering part, some other filters should be tried in order to try to minimize the error coming from the outliers without decreasing the quality of the signal. This would allow for a more accurate estimation of the model and also it could be useful in order to estimate the load and position, because the impedance values to be compared are taken from the filtered signal and not from the raw data. So, if applying a better filter would return a smoother signal it would return less errors due to the outliers.

- Modeling

Concerning the modeling method, other different estimation methods could be implemented to be able to contrast the results. Methods developed for one dimensional continuous signals could prove more useful than the one implemented in this work or at least they could be used to see if they report any differences when using them to perform the load and position estimation.

Another option that could be tried would be to change the initialization of the problem, looking for an initialization closer to the expected results that could improve the optimization when it starts.

### 5.4.2 Analysis of heating phase

Apart from the improvements that could be obtained from the modeling, it would also be interesting to analyze the rise of the impedance between heating time and cooling time, since the SMA behaves differently in that area. It would also be a question of performing a similar analysis to the one proposed in this project between those times looking for the cost function that most resembled the function in that range.

### 5.4.3 Changing electric current

Besides, as it has been explained at the beginning of the work, the change of shape and phase in the SMA has been obtained by applying an electric current, so, by changing this current, more shifting could be explored that could help understand better the behavior of this material. So, another area for improvements could be to analyze more in detail the variations caused by this electric current applied when studying its position and load.



## 6 Budget

The total cost generated by the project is broken down into two sections: the cost related to the personnel (shown in Table 6.1) and the cost generated by the materials and equipment needed for the development of the project (shown in Table 6.3) and the budget corresponding to the software license needed to use MATLAB 6.4. These three tables show the approximate cost of each of the items corresponding finally to the direct costs generated by the project.

Even though in the project the work is not performed directly on them, for the obtaining of the signals the following materials shown in Table 6.2 were necessary: SMA wire and the sensors to measure position, so they should be included in the project budget because without them nothing could have been done.

Personnel	Hours	Salary (€/hour)	Total (€)
Biomedical Engineer	650.00	20.00	13,000.00
Project Coordinator	70.00	45.00	3,150.00
<b>Total</b>			<b>16,150.00</b>

**Table 6.1:** Personnel budget.

Material	Quantity (m)	Cost/Quantity (€/hour)	Total (€)
SMA wire	75	6.00	450.00
<b>Total</b>			<b>450.00</b>

**Table 6.2:** Materials budget.

To compute the total cost of the equipment and the software the following formula was applied:

$$Total\ cost = \frac{Devoted\ time}{Depreciation\ period} \times Cost \quad (6.1)$$

To estimate the depreciation period of the equipment used, several sources were investigated. For the micro-controller, it was found that it had a longevity of 10 years [28]. This was the same result observed when looking for an approximation for the rest of the equipment [29].

Equipment	Cost	Devoted time (months)	Depreciation period (months)	Total (€)
Personal Computer	600.00	7	60	70.00
Position sensor (NSE5310)	30.00	7	120	1.75
Micro-controller (STM32F4)	17.00	7	120	0.99
Current sensor (Allegro MicroSystems acs722)	5.00	7	120	0.29
(Power Supply)	50.00	7	120	2.92
			<b>Total</b>	<b>75.95</b>

**Table 6.3:** Equipment budget.

The software used for the project is MATLAB. They offer the option of purchasing an annual license or a perpetual one. For this project an annual license would have been enough, but if this software is going to be used in the future, then the best option would be to purchase the perpetual license. So, the cost added to Table 6.4 is the one corresponding to the perpetual license, and then the equation 6.1 is applied because that license can be exploited over time. The depreciation time will be set to 3 years (36 months), because it is the typical time that the vendor provides technical support and software updates to the user [30].

Software	Cost	Devoted time (months)	Depreciation period (months)	Total (€)
MATLAB	2,000.00	7	36	388.89
			<b>Total</b>	<b>388.89</b>

**Table 6.4:** Software budget.

Finally, the total cost of the project is calculated by taking into account the previously stated direct cost, and adding to this number, the indirect cost to cover utilities and shipping costs and is calculated as 20% of the direct cost, and the 21% value added tax (VAT) in Spain is added as well to the final cost of the project which is shown in Table 6.5.

Description		Cost (€)
Direct Cost	Personnel	16,150.00
	Equipment	75.95
	Software	388.89
Indirect Cost		3,412.97
VAT		4,300.34
<b>Total Cost</b>		<b>24,778.15</b>

**Table 6.5:** Total cost.

After all the calculations are performed, it can be seen that the total cost of the project adds up to a value of **24,778.15 €**.

## Annex A: Acronyms

- **AEMPS:** Agencia Española de Medicamentos y Productos Sanitarios
- **DE:** Differential Evolution
- **DSC:** Differential Scanning Calorimeter
- **EA:** Evolutionary Algorithm
- **EAP:** Electroactive Polymer
- **SMA:** Shape Memory Alloy
- **SME:** Shape Memory Effect
- **SMP:** Shape Memory Polymer
- **SSD:** Sum of Squared Differences
- **VAT:** Value Added Tax

## Bibliography

- [1] H. Qian, H. Li, G. Song, and W. Guo. Recentering shape memory alloy passive damper for structural vibration control. *Mathematical Problems in Engineering*, 2013:1–13, 11 2013.
- [2] Nickel Titanium. [https://en.wikipedia.org/wiki/Nickel\\_titanium](https://en.wikipedia.org/wiki/Nickel_titanium). [Last access: July 2018].
- [3] SMARTWIRES. Nitinol - Frequently asked questions. [https://smartwires.eu/index.php?id\\_cms=9&controller=cms&id\\_lang=1](https://smartwires.eu/index.php?id_cms=9&controller=cms&id_lang=1). [Last access: August 2018].
- [4] L. Lecce and A. Concilio. *Shape Memory Alloy Engineering*, chapter 1.3. Shape Memory Alloys. Butterworth-Heinemann, 09 2014.
- [5] C. Wen *et al.* Mechanical behaviors and biomedical applications of shape memory materials: A review. *AIMS Journal*, 5, 06 2018.
- [6] eSMART. Robotic muscles. [http://webdocs.cs.ualberta.ca/~database/MEMS/sma\\_mems/muscle.html](http://webdocs.cs.ualberta.ca/~database/MEMS/sma_mems/muscle.html). [Last access: August 2018].
- [7] S. Kalra, B. Bhattacharya, and B.S. Munjal. Design of shape memory alloy actuated intelligent parabolic antenna for space applications. *Smart Materials and Structures*, 26(9):095015, 2017.
- [8] Measuring Transformation Temperatures in Nitinol Alloys. <http://jmmedical.com/resources/211/Measuring-Transformation-Temperatures-in-Nitinol-Alloys.html>. [Last access: August 2018].
- [9] Orthosis. <https://medical-dictionary.thefreedictionary.com/orthosis>. [Last access: August 2018].
- [10] S. Dilibal, E. Guner, and N. Akturk. Three-finger SMA robot hand and its practical analysis. *Robotica*, 20, 01 2002.
- [11] Global Legal Research Center The Law Library of Congress. Approval of Medical Devices. <https://www.loc.gov/law/help/medical-devices/approval-of-medical-devices.pdf>, 09 2014.

- [12] D.J. Fernandes, R.V. Peres, A.M. Mendes, and C.N. Elias. Understanding the shape-memory alloys used in orthodontics. *ISRN Dentistry*, 2011:1–6, 2011.
- [13] M. Santoro, O.F. Nicolay, and T.J. Cangialosi. Pseudoelasticity and thermoelasticity of nickeltitanium alloys a clinically oriented review. part i temperature transitional ranges. *American Journal of Orthodontics and Dentofacial Orthopedics*, 2001.
- [14] Hysteresis. <https://www.dictionary.com/browse/hysteresis>. [Last access: August 2018].
- [15] Ľ. Miková, S. Medvecká-Beňová, M. Kelemen, F. Trebuňa, and I. Virgala. Application of shape memory alloy (SMA) as actuator. *Metalurgija*, 54(1):169–172, 2015.
- [16] S. Pittaccio, L. Garavaglia, C. Ceriotti, and F. Passaretti. Applications of shape memory alloys for neurology and neuromuscular rehabilitation. *Journal of Functional Biomaterials*, 6(2):328–344, 06 2015.
- [17] T. Tang, D. Zhang, T. Xie, and X. Zhu. An exoskeleton system for hand rehabilitation driven by shape memory alloy. *Proceedings of 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 756–761, 12 2013.
- [18] Ashwin Rao, A.R. Srinivasa, and J.N. Reddyht. *Design of Shape Memory Alloy (SMA) Actuators*, chapter 1: Introduction to Shape Memory Alloys, pages 1–31. Springer, 2015.
- [19] W. Rottiers, L. Van den Broeck, C. Peeters, and P. Arras. Shape memory materials and their applications. 2011.
- [20] T. Mirfakhrai, J.D.W. Madden, and R.H. Baughman. Polymer artificial muscles. *Materials Today*, 10(4):30 – 38, 2007.
- [21] K. Takashima, J. Rossiter, and T. Mukai. Mckibben artificial muscle using shape-memory polymer. *Sensors and Actuators A: Physical*, 164(1):116 – 124, 2010.
- [22] S. Herranz Moreiro. Control de posición de un material con memoria de forma (SMA). Universidad Carlos III de Madrid. [https://e-archivo.uc3m.es/bitstream/handle/10016/11642/PFC\\_Saul\\_Herranz\\_Moreiro.PDF;jsessionid=9A658FD7F5937A0469D30B7A214FE5F0?sequence=1](https://e-archivo.uc3m.es/bitstream/handle/10016/11642/PFC_Saul_Herranz_Moreiro.PDF;jsessionid=9A658FD7F5937A0469D30B7A214FE5F0?sequence=1), 2011.

- [23] K. Price and R. Storn. Differential Evolution (DE). <http://www1.icsi.berkeley.edu/~storn/code.html>. [Last access: August 2018].
- [24] Introduction to Evolutionary Algorithms. <https://towardsdatascience.com/introduction-to-evolutionary-algorithms-a8594b484ac>. [Last access: July 2018].
- [25] A.P. Engelbrecht. *Computational Intelligence: An Introduction*, chapter 13: Differential Evolution, pages 237–260. Wiley, 2011.
- [26] V. Arunachalam. Optimization using differential evolution. *The University Of Western Ontario. Department of Civil and Environmental Engineering.*, 2008.
- [27] Differential Evolution. [https://en.wikipedia.org/wiki/Differential\\_evolution](https://en.wikipedia.org/wiki/Differential_evolution). [Last access: July 2018].
- [28] Product Longevity. [https://www.st.com/content/st\\_com/en/support/resources/product-longevity.html](https://www.st.com/content/st_com/en/support/resources/product-longevity.html). [Last access: September 2018].
- [29] University of California: Equipment useful life table for depreciation. [https://www.ucop.edu/procurement-services/\\_files/euliddesc2000.pdf](https://www.ucop.edu/procurement-services/_files/euliddesc2000.pdf). [Last access: September 2018].
- [30] Perpetual software license. <https://whatis.techtarget.com/definition/perpetual-software-license>. [Last acces: September 2018].