Universidade de Aveiro Departamento de Engenharia Mecânica
2017

Renato Miguel
Emílio Rocha

Modelação de gerador eletromagnético com redes neuronais artificiais

Modelling of an electromagnetic generator using Artificial Neural Networks

Universidade de Aveiro Departamento de Engenharia Mecânica
2017

Renato Miguel
Emílio Rocha

Modelação de gerador eletromagnético com redes neuronais artificiais

Modelling of an electromagnetic generator using Artificial Neural Networks

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requesitos necessários à obtenção do grau de Mestre em Engenharia Mecânica, realizada sob a orientação científica de Marco Paulo Soares dos Santos, Professor Auxiliar Convidado do Departamento de Engenharia Mecanica da Universidade de Aveiro e Jorge Augusto Fernandes Ferreira, Professor Auxiliar do Departamento de Engenharia Mecânica da Universidade de Aveiro.

Apoio financeiro dos projetos
UID/EMS/00481/2013-FCT e CENTRO-01-0145-FEDER-022083

o júri / the jury

presidente / president	**Professor Doutor Vítor Manuel Ferreira dos Santos** Professor Associado da Universidade de Aveiro
vogais / examiners committee	**Professor Doutor Carlos Alberto da Costa Bastos** Professor Auxiliar da Universidade de Aveiro
	Doutor Marco Paulo Soares dos Santos Professor Auxiliar Convidado da Universidade de Aveiro (orientador)

agradecimentos / acknowledgements

Em especial aos meus pais, avós, irmão e namorada pelo apoio incondicional. Aos meus amigos por me acompanharem ao longo deste percurso. Aos meus colegas do LAR por toda a ajuda preciosa. Aos professores Marco Paulo Soares dos Santos e Jorge Augusto Fernandes Ferreira pela disponibilidade, empenho e motivação. Ao Engenheiro António José de Fonseca Festas e ao Professor António Manuel de Amaral Monteiro Ramos pelo auxílio no projeto mecânico e todos os outros que me ajudaram ao longo deste projeto.

palavras-chave

Energy Harvesting, Harvester modeling, Auto-geração, Redes Neuronais Artificiais, NARX

resumo

A crescente necessidade de sistemas de funcionamento autónomo aliada à redução de consumo por parte dos dispositivos microeletrónicos ao longo dos últimos anos, tem motivado a investigação de dispositivos para auto geração. O desenvolvimento de um dispositivo para *energy harvesting*, considerando uma determinada aplicação, requer o seu estudo e otimização. Consequentemente, a modelação do sistema para efeitos de simulação torna-se imperativa. A utilização de modelos matemáticos analíticos ou FEM é uma abordagem standard no desenvolvimento de um modelo para computação. No entanto, estas abordagens apresentam-se morosas, devido às restições temporais estabelecidas não só pelo desenvolvimento do modelo, mas também pela sua simulação. Neste trabalho, a aplicação de Redes Neuronais Artificiais para a modelação da dinâmica de um *harvester* baseado em levitação magnética é investigada. A recolha de dados requerida pela metodologia das Redes Neuronais Artificiais impôs o desenvolvimento de um dispositivo adequado para a aquisição de dados intrínsecos ao sistema. Uma estação de testes foi construída com o objetivo de induzir excitações rotacionais no dispositivo e adquirir a dinâmica de movimento mecânico dos ímanes em levitação. Diferentes arquiteturas de redes e técnicas de implementação são abordadas neste trabalho, de modo a otimizar as características do modelo. Das diferentes abordagens tidas para implementação de um modelo de redes neuronais, a configuração denominada neste trabalho como NARX BR_{OC} permitiu a obtenção de correlações superiores a 95% para simulações dentro e fora da gama de treino, quando comparadas com resultados experimentais. O desempenho do gerador desenvolvido é também analisado e discutido de acordo com aplicações pretendidas

keywords Energy Harvesting, Harvester modeling, self-powering, Artificial Neural Networks, NARX

abstract The increasing necessity for autonomous functioning systems alied with comsumption reduction by microelectronic devices over the last years, has motivated the research on self-powering devices for remote applications. Developing an energy harvestring device for a determined application requires its study and optimization. Therefore, modeling the dynamics of the system for simulation purposes becomes mandatory. The use of analytical mathematical models or FEM is a standard approach for the development of computable models. However, this approach reveals to be time-consuming due to temporal restrictions established not only by the model development but also by its simulation. This work investigates the application of Artificial Neural Networks on the modeling of magnetic levitation systems for energy harvesting purposes. The data collection implied by a Neural Network approach demanded the development of a device suitable for the acquirement of information intrinsic to the system. A testing station was built with the goal to induce rotational excitations on the device and acquire the motion dynamics of the levitation magnet. Different network architectures and implementation techniques are approached in this work in order to optimize the characteristics of the model. From the different approaches taken for proper model implementation, the configuration named in this work as NARX BR_{OC} allowed the attainment of correlation values above 95% for simulations inside and outside the training range, when compared with experimental results. Also, the performance of the developped generator is analyzed and discussed according to intended applications

Contents

List of Figures

List of Tables

Nomenclature

Acronyms

ANN	Artificial Neural Network
CAD	Computer-aided design
CC	Cross-Correlation Coefficient
emf	electromotive force
FEM	Finite Elements Method
HLN	Hiddel Layer Neurons
LBEH	Levitation Based Energy Harvesters
MPE	Mean percentage error
MSE	Mean square error
NARX	Nonlinear Autoregressive Network with Exogenous Inputs
TDL	Tapped Delay Lines

Symbols

α	Eletromagnetic couple coefficient [T.m]
β	Excitation Wave Angle Amplitude [degrees]
$\mathbf{m}$	Magnetic moment $[A.m^2]$
δ_{MC}	Distance from moving magnet to coil $[m]$
$\delta_{t/b}$	Distance from moving magnet to top/bottom magnet $[m]$
$\dot{x}, \ddot{x}$	Time derivate with order equal to number of points
$\hat{n}$	Normal Vector
μ_0	Free space permeability $[H/m]$
ω	Angular frequency $[\frac{rad}{s}]$
Φ	Magnetic flux through coil $[Wb]$
θ	Excitation Wave Angle [degrees]

ε Eletromotive induced force [V]

φ Excitation Wave Delay [s]

$\vec{A}$ Magnetic Potential Vector [$\frac{V.s}{m}$]

B Magnetic Flux Density [T]

B_e External magnetostatic field [T]

B_r Residual Magnetic Flux [T]

c Total damping coefficient [$\frac{N.s}{m}$]

c_a Viscous damping coefficient [$\frac{N.s}{m}$]

c_e Magnetic damping coefficient [$\frac{N.s}{m}$]

c_p Dry friction damping coefficient [$\frac{N.s}{m}$]

d_M Magnet's diameter [m]

d_{C_i} Coil inner diameter [m]

d_{C_o} Coil outter diameter [m]

d_{H_i} Harvester's inner diameter [m]

d_{H_o} Harvester's outter diameter [m]

E Eletrical Field [V/m]

f Frequency [Hz]

F_e Magnetic damping force (Lorentz Force) [N]

F_{Fric} Dry friction damping force [N]

$F_{mag_{t/b}}$ Magnetic repulsive forces by top/bottom magnet on levitating magnet [N]

g Gravity acceleration

H Magnetic field Strength [A/m]

h_M Magnet's height [m]

h_C Coil height [m]

h_H Harvester's height (distance between fixed magnets) [m]

i Eletrical current

k_{mag} Magnetic spring coefficient [$\frac{N}{m}$]

L Indutance [H]

M Magnetization [A/m]

m Levitation mass [kg]

N	Number of loops in coil
P	Excitation Wave Period [s]
R	Resistence [Ω]
t	Time [s]
U	Magnet's potential energy [J]
u', u''	Space (x) derivate with order equal to number of points
V	Voltage

Chapter 1

Introduction

1.1 Context

In the course of the last four decades the development of low consumption electronic devices has been a field of huge technological advancements. Those advances allied with a continuous evolution of wireless communications and a progressive size reduction of electronic devices in general, has allowed an application range enlargement for autonomous systems for data acquirement and transmission [1]. As a result of this technological progress the interest in developing an equally self-employed and renewable energy source led the academic community to resort to Energy Harvesting technologies. This concept, also known as energy scavenging, represents a process from which it is possible to draw electrical energy from a certain environment, usually for low consumption applications.

The wide range of applicable fields for this energy generation technique motivates the investigation of transduction systems for several environments with distinct characteristics and energy sources where there is the need to power electronic circuits or other components, whether being for data reading or transmission, manipulating a physical variable or any other low power application [2].

From the wide variety of potential electrical energy sources, a considerable number of methods to obtain it arises, multiplying the number of devices and configurations already developed and the possibility for new ones to be engineered. While Solar, Chemical and Thermal power are considered viable sources, the harnessing of energy from natural or artificial motion has been recognized as one of the most applicable energy sources for the transduction of electric power [3, 4].

The harness of electrical energy from motion is mainly accomplished through one of the three transduction principles, piezoelectricity, electrostatics or electromagnetic induction. All three present inherent advantages and disadvantages, however, piezoelectric and electromagnetic generators have been receiving special attention from the academic community, particularly because of their higher energy density, easy application and reduced maintenance requirements [5, 6].

The design and conception of a motion energy transduction system must be performed not only to take advantage of available mechanical energy in the surrounding environment, but also in order to reduce the need of human intervention whether for maintenance or any other kind of manipulation [7, 8]. Features like functional and manufacturing simplicity, reliability, low maintenance requirement and low intermittence are key topics when approaching the development of an energy harvesting device.

Magnetic levitation based harvesters use the electromagnetic induction principle for motion to electricity transduction. The varying magnetic flux through a coil induces an electromotive force on the coil which results in a current when the circuit is closed with a load resistance. The

electromagnetic induction allied with a magnetic levitation architecture has been proving its versatility and reliability in numerous energy transduction applications. This induction mechanism presents a simple functioning principle as well as good efficiency and low degradation, making this an excellent transduction method to develop a low cost, highly autonomous and long lasting harvester device (figure 1.1). In addition, the replacement of a mechanical transmission system for a magnetic levitation system allows the removal of components like springs or other parts used to transfer inertial energy, diminishing losses through mechanical dissipation.

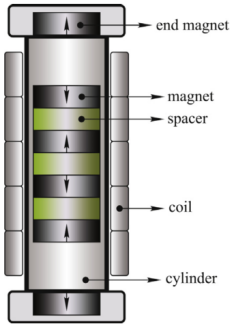


Figure 1.1: LBEH with electromagnetic transduction principle [9]

1.2 Objectives

The proposed work for this thesis consists in the design, construction and modeling of the dynamics for a motion-driven LBEH generator prototype. The mathematical modeling though Analytical, Semi-Analytical, Numerical or Empirical analysis has been the main approach taken in the literature, however, modeling the architecture of an energy harvester is usually a cumbersome and time consuming process, mainly when high accuracy is demanded. Machine Learning tools like *Artificial Neural Networks*(ANN) are known for the ability of emulating non-linear dynamic systems over time, such as the proposed generator [10, 11].

Having ANN mathematical modeling techniques into account, and considering the purpose of this prototype its dynamic analysis, for subsequent dimensional and functional optimization, the mining for the maximum amount of data regarding its performance becomes a requirement. The acquired data aims for the creation of an output database, function of the behavior of the generator under distinct external conditions, making it possible to train a neural network structure.

The necessity for data mining implies the development of a testing platform, capable of emulating the stimulation that would be applied to the generator in its potential working environment and consequently the acquisition module able to read and save the data extracted from the functioning process of the generator when the excitation is applied.

In this way, the main objectives for this thesis may be summarized in the following topics:

- Development and construction of a generator prototype for the transduction of motion energy into electricity.
- Development and programming of a test bench for the generator, including acquisition module and actuator.
- Structuring of a neural network able to replicate the dynamics of the device.
- Acquired data implementation and network training for prediction simulations.

Chapter 2

Literature review

2.1 LBEH concepts

Harvesting energy from the motion of artificial or natural environments has been showing great potential for a wide range of applications. The existence of several applicable transduction methods and the vast number of possible sources makes this a very rich and diversified area of investigation concerning architectures of harvesters. However, favorable features provided by electromagnetic levitation harvesters makes it a very important subject of study for the academical community.

According to reviewed literature, magnetic levitation harvesters possess rather similar structural configurations. It often consists in a levitating case on which a free magnet levitates due to magnetic repulsion, as can be seen in figure 2.1. Cylindrical harvesters are the most common structural shape presented on the literature, however, other parameters and aspects are proposed in several publications[12, 13].

A standard configuration by Soares dos Santos et al [5], in which a small dimension cylindrical harvester is presented for low consumption applications. Figure 2.1 shows the proposed generator on the referred article.

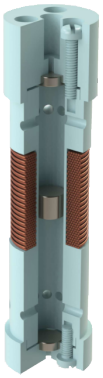


Figure 2.1: Proposed magnetic levitation harvester by Soares dos Santos et al [5]

While assuming a classical cylindrical configuration, other generators are presented with various enhancement features or just dimensional variations. Mann and Owens [2] propose a rather similar device (figure 2.2), however, adding a guiding wire for friction reduction and external magnets, provide the harvester a bistable potential well, thus increasing its nonlinear properties.

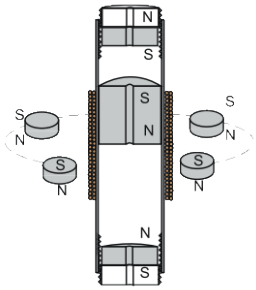


Figure 2.2: LBEH with guiding wire and external magnets proposed by B.P.Mann et al. [2]

Nammari et al [12] also propose a device with similar configuration, that combines the non-linearities of the magnetic forces with mechanical springs linear properties in order to evaluate its behavior for a wide range of mechanical and magnetic stiffness ratios (figure 2.3).

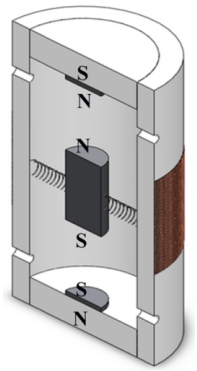


Figure 2.3: Electromagnetic levitation harvester with magnetic and mechanic spring combination [12]

Other approaches take into consideration heavier mass systems for lower frequency applications, where despite architectures similar to the previously presented, dimensional enlargement is considered in order to provide more inertia to the system [14, 15].

Outside cylindrical configurations, Berdy et al. [6] proposed a block shaped harvester in order to allow a thinner device. The overall structure and working principles are analogous to the ones presented previously, in this case, disk or ring magnets were replaced by parallelepiped magnets, as can be seen in figure 2.4

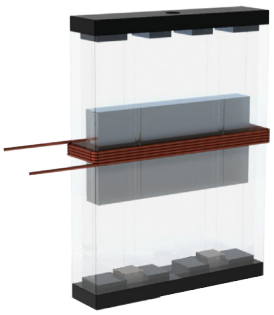


Figure 2.4: Box shaped magnetic levitation harvester proposed by Berdy et al. [6]

2.2 LBEH modeling

The development of the governing mathematical equations for a LBEH is one of the most important steps in its development and optimization. The dynamic simulation, and the transient analysis of the system provide important information for dimensional tuning. For a better comprehension of the various interpretations made by each author, the equations have been rewritten in order to comply an homogeneous variable system presented in the nomenclature. It has also been considered a global referential system, so the analysis made in the different articles can be correctly interpreted. The adopted referential system can be seen in figure 2.5.

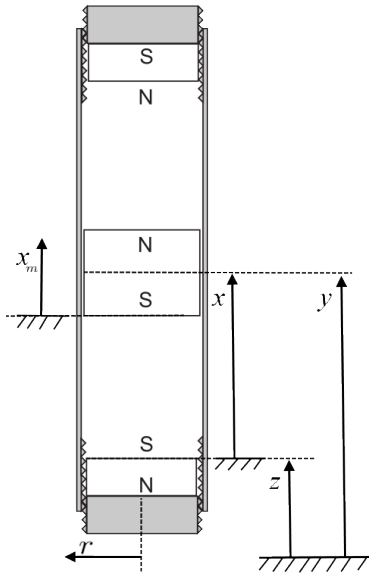


Figure 2.5: Referential structure used for modeling analysis

Although power outputs are often modeled and computed, this review focuses on the analysis of parameters that play a direct role on the motion differential equations. Those variables are the following:

- **Magnetic Field of stationary and levitating magnets :** Magnetic Field density emanated from both constrained and free magnets

- **Interaction between magnetic forces:** Repulsive forces between two or more magnets

- **Induced eletromotive force:** Induced eletromotive force resultant from the relative motion between coil and magnet

- **Coil induced current:** Eletrical current generated by the eletromotive force induced in the coil

- **Friction and Damping Forces:** Forces resultant from mechanical contacts, Lorentz force or air flow resistance

In the literature the described topics are often approached using 1 of 4 different mathematical methodologies, Analytical, Semi-Analytical, using Numerical techniques like FEM or using Empirical approaches [2, 5, 9, 16].

Analyzing the literature it is possible to conclude that there are two main methodologies concerning the dynamic modeling for a levitation system. Most authors opt for the classical second order differential spring damper mass equation, which is a trustworthy method to model this kind of mechanical systems [4, 5, 6, 9, 12, 14, 15, 17, 18]. On the other hand, some authors prefer to analyze the frequency response of the levitating magnet using the Multiple Scale technique to approximate the solution of the perturbation problem [16, 19]. For the differential analysis, the canonic equation for system representation is generally set as $-m\ddot{z} = m\ddot{x} + c\dot{x} + k_{mag}x$ [4]. This equation expresses the response of the magnet when an external excitation is applied to the LBEH body. In this type of analysis, the biggest difference studies comes from the way that the system variables modeling is performed. Also, some authors also take into consideration the dry friction forces, F_{fric} that actuate on the LBEH system.

Magnetic Field

Magnetic field modeling is required for a well developed model. This is due to its influence on all the electromagnetic system properties, from magnetic repulsive forces, to emf, induced current and the resultant magnetic damping force. Proper definition of the magnetic field provides a more reliable model of a LBEH system. The majority of the reviewed literature includes this step in their analysis, mostly using analytical and FEM techniques. Both Coulomb and Equivalent Surface Current models have been used by authors to obtain the magnetic flux density as a function of radial and axial distances or of a 3 dimensional coordinate. Berdy et al.[6] used the Coulombian model [20] to calculate the z-directed flux at an arbitrary point in a three-dimensional coordinate system for a block magnet.

The Equivalent Surface Current [21] model is also used in the literature to model the magnetic flux density as a function of z and r to the center of the magnet. In this analysis the author discretized the magnet into a finite set of current loop elements and then superimposes the magnetic fields of the constituent current loops to obtain the magnetic flux density [5].

In numerical approaches to this problem, the FEM is prominent amongst studies [4, 22]. Saravia et al. [9] used this technique to compute the magnetic field on a LBEH magnet array and respective fixed magnets, as well as the influence on each other. Figure 2.6, shows an example of the magnetic field FEM computation result. Wang et al. [14] also use this technique to tackle the problem, the authors focus on the effect of magnetic pole inversion on the magnetic flux density of a magnet array, figure 2.7.

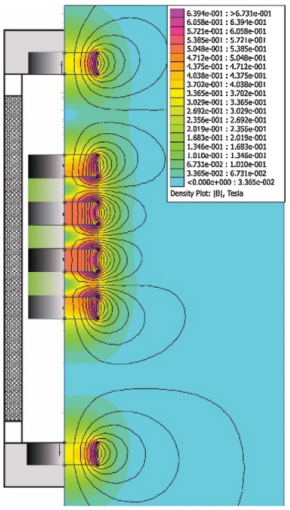


Figure 2.6: FEM simulations for magnetic flux density distribution [9]

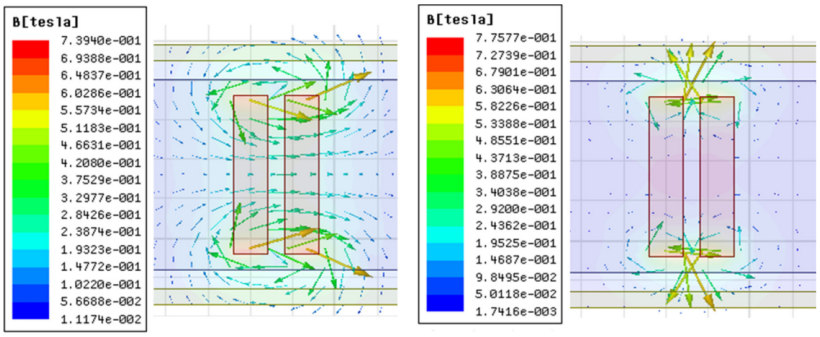


Figure 2.7: FEM simulations for magnetic flux density distribution in two pole configurations (a) opposite poles facing each other (b) same pole facing each other [14]

Interaction Magnetic Forces

For the magnetic levitation and restoring forces modeling, several different approaches are taken in the literature. One of the most common is an Numerical/Empirical method that consists in the fitting of real measurements (figure 2.8) into a power series . Mann et al. as well as other authors [9, 14, 15, 19] use this approach with practical measured values. It is most often used a polynomial with order 3 and 1 terms, $k_1 x + k_3 x^3$.

Considering purely Analytical approaches, Bernal et al [16] considered the equation based on the Coulombian model, where the force exerted by an external magnetic field B_e on 2D magnets with magnetization M is given by equation 2.1.

$$F = -\oint_S (M.\hat{n}) B_e(x) d^2x, \tag{2.1}$$

Considering the magnets interacting surfaces, the authors then obtains a derivable polar coordinate expression 2.2, from which its possible to compute force values recurring to Bessel integral functions of order one.

$$F_z = M\left(\int_0^{d_M/2}\int_0^{2\pi} B_e(x) d\theta \rho d\theta\Big|_{x=0} - \int_0^{d_M/2}\int_0^{2\pi} B_e(x) d\theta \rho d\theta\Big|_{x=-h_M}\right) \tag{2.2}$$

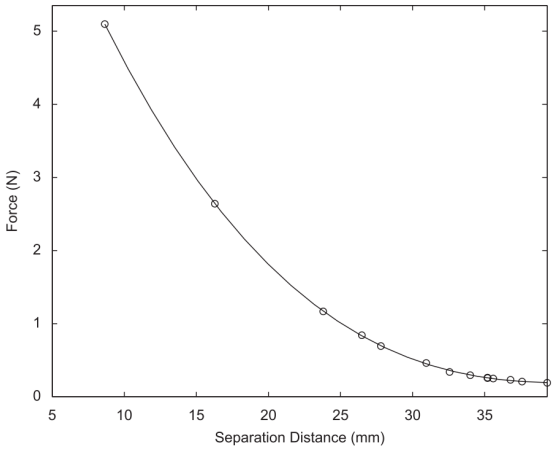


Figure 2.8: Measured values for curve fitting [19]

Nammari et al.[12] take a hybrid approach when analytically computing the magnetic forces deriving the interaction energy between magnets recurring to the Bessel functions [5], then applying the previously referred polynomial fitting technique to the obtained values in order to enhance computational efficiency with a simpler accurate model.

Electromotive Induced Force

This variable is most frequently derived from Faraday's Law of induction. However, some more exhaustive approaches use the Ampère's Circuital law [4] or equations like the Maxwell-Faraday one for the modeling of electromotive induced force. Another variable parameter is the coils configuration itself. Some Semi-Analytical analysis take into consideration a discrete coil made of individual circular rings [5] or even as a solid cylindrical tube with dimensions similar to the ones of the real coil [17].

From the Faraday's Law of induction, many computable equations are obtained, resulting from different approaches and derivations of it. Some authors apply it almost directly, considering a voltage proportional to the speed of the magnet, $\varepsilon = \alpha \frac{dx}{dt}$, in which the proportion term is the magnetic coupling coefficient [6]. In studies like the one published by Kecik et al. [15], a dynamic coupling coefficient is modeled, this model takes into consideration the variable couple coefficient effect in the induced emf and also its effect on the dynamics of the device.

Other authors like Saravia et al [9] apply this Law to use in a discrete analysis, where it is possible to obtain the summation of the flux in every loop of the coil as can be seen in equation 2.3.

$$\varepsilon = -\Big(\sum_{i=1}^{nc} \int_S B'(x_i)\, dAi\Big)\dot{x} = -\sum_{i=1}^{nc} \Phi' \dot{x}, \tag{2.3}$$

In a similar approach to the one proposed by Saravia et al [9], Soares dos Santos et al [5] makes use of the Maxwell-Faraday equation to model the output voltage in the coil generated by the time-varying magnetic flux that permeates it. The author takes into consideration a coil composed of a set of singular circular turns and a 3D surface bounded by a closed contour defined by each turn. From the original Maxwell-Faraday equation for induction 2.4

$$\nabla \times E = -\frac{\partial B(t)}{\partial t}, \tag{2.4}$$

a computable equation is obtained using the Kelvin-Stokes theorem and considering an uni-

form spatial distribution along coils,

$$V = 2\pi \frac{\partial}{\partial t}\Big(\sum_{k=1}^{N_z} \sum_{j=1}^{N_r} \int_0^{r_j} B_z(r, z_k) r\, dr \Big) \tag{2.5}$$

Coil induced current

A standard framework to obtain the current induced in the coil is to directly apply Ohm's Law, $V = R_{Total}.I$, with the dynamic electromotive force induced in the circuit. This approach is most times done taking into account voltage as, $V = \alpha.\dot{x}$, but it is mostly accurately computed when considering also a dynamic coupling coefficient [6, 12]. Berdy et al. took this aspect into consideration in their article and simulated current values using equation 2.6.

$$i(t) = \frac{\alpha(x(t))\dot{x}}{R_{Load} + R_{Coil}} \tag{2.6}$$

More complete approaches account not only the resistive load of the system but also inductive loads [2, 5, 16]. Relating the electromotive induced force on the coil with the impedance through differential equations, Mann et al. formulated the problem using equation 2.7. Analogous approaches taken in related studies [5, 16, 9] use similar equations.

$$L\dot{i} + iR + \alpha\dot{x} = 0 \tag{2.7}$$

Friction and Damping Forces

One of the most important set of variables to model when looking to accurately simulate the dynamics of an harvester are the friction and damping forces. The Karnopp friction model was used in the literature to model the friction forces between moving magnet and the inner surface of the device [5]. This model considers the speed of the magnet to compute the referred forces.

$$F_{fr} = \begin{cases} f_{re} & \text{if } -f_{bw_n} < f_{re} < f_{bw_p} \\ f_{co_p} + k_{v_p}\dfrac{dx}{dt} & \text{if } \frac{dx}{dt} > v_{min} \\ -f_{co_n} + k_{v_n}\dfrac{dx}{dt} & \text{if } \frac{dx}{dt} < -v_{min} \end{cases} \tag{2.8}$$

Where,

$$f_{re} = m\ddot{x} - mg - F_{mag_t} + F_{mag_b} - F_e \tag{2.9}$$

While the force resultant on the magnet lies between the break-away interval ($f_{bw_n} < f_{re} < f_{bw_p}$), friction equals to f_{re}.

After distinguishing those force limits, the model defines two different friction conditions for each movement direction, in which f_{co_n} and f_{co_p} represent the Coulomb friction forces for positive and negative speeds respectively, while k_{v_n} and k_{v_p} represent the damping coefficients for the same conditions.

Damping forces are approached in the literature mainly accounting for two main principles: the viscous damping and the electromagnetic damping. While viscous damping is always considered according to static models [4, 5, 19], the damping resultant from Lorentz Force on the coil has strong dynamic features. Berdy et al. [6], consider a position dependent electromagnetic damping coefficient according to the equation 2.10.

$$c_e(x(t)) = \frac{\alpha^2(x(t))}{R_{Load} + R_{Coil}} \tag{2.10}$$

For compactness sake, a qualitative analysis to several reviewed studies on the subject is presented in tables 2.1 and 2.2.

Author[Article]	Magnetic Field	Interaction Magnetic Forces	Eletromotive Induced Force	Coil Induced Current	Friction and Damping Forces
B.P.Mann et al. [19]	-	**Empirical/Numerical Method** Real Measurements fitted into a power series $F_{mag}(x) = \Sigma_{n=0}^{3} \alpha_n x^n$	-	**Analytical Method** Kirchoff Law applied to the generator circuit $i = \alpha \frac{\dot{x}}{R_{Load}+R_{Coil}}$	**Analytical Method** Static damping coeficient c, where $c = c_p + c_e$ and $c_e = \frac{\alpha^2}{R_{load}+R_{int}}$
B.P.Mann et al. [2]	**Analytical Method** Magnetic field calculated at a location r_p due to a magnet located at r_s is $B = -\frac{\mu_0}{4\pi} \nabla \frac{\mathbf{m}_s . r_{p/s}}{\lvert r_{p/s} \rvert^3}$	**Analytical Method** Magnetic interaction forces calculated from the potential energy of a magnet at r_p in the field generated by a magnet at r_s is $U = -\mathbf{m}_p . B$	-	**Differential Analysis** Use of a first order differential equation to relate dynamics and current output $L\dot{i} + iR + \alpha\dot{x} = 0$	-
D.F.Berdy et al [6]	**Analytical Method** Use of Coulombian model to calculate magnetic flux density at point (i, j, k) for block shaped magnet	**Analytical Method** Use of Coulombian model to calculate force between two rectangular magnets	**Analytical Method** Use of Faraday's law of induction to calculate the emf induced on the coil $\varepsilon = \alpha \frac{dx}{dt}$	**Analytical Method** Ohm's Law applied to circuit with dynamic magnetic coupling coeficient $i(t) = \frac{\alpha(x(t))\dot{x}}{R_{Load}+R_{Coil}}$	**Analytical Method** Consideration of static $F_{Fric} = \frac{\dot{x}(t)}{\text{mod}\,\dot{x}(t)} c_p$ and dynamic $F_e = c_e(z(t))\dot{z}(t)$
A.R.Foisal et al [4]	**Numerical Method** Magnetic flux density in a coil produced by a magnet at a distance of d_{MC}	**Analytical/Numerical Method** Resultant magnetic force equations set as $F = \frac{\mu_0 H_m}{4\pi} \left[\frac{H_t}{\delta_t^2} - \frac{H_b}{\delta_b^2} \right] = k_{mag} x$ and estimation of k_{mag} using the linear region of simulation	**Analytical Method** Reference to Ampére's Circuital Law with voltage across coil given by $\varepsilon = -BL_p\dot{x}$	-	**Analytical Method** Static total damping coefficient $c = c_p + c_e$ with $c_e = \frac{(N.B)^2}{R_{Load}+R_{Coil}+j\omega L_{Coil}}$
A.G.Bernal et al [16]	**Analytical/Numerical Method** Use of the magnetic scalar potential to obtain the magnetic field density equation and further comparison with FEM model	**Analytical Method** Single Magnet force equations obtained from the integration of the surface term of magnetostatic field $F = \oint_S (M.\hat{n}) B_e(x) d^2x$	**Analytical Method** Integration of the the emf on the coil from Faraday's law in integral form $\oint_C E.dl = -\int_S \frac{\partial B}{\partial t} d^2x$	**Differential Analysis** Induced current relation with emf equation set as $\varepsilon = (R_{Coil} + R_{Load})i + L_{Coil}\dot{i}$	-
M.L.Morgado et al [17]	**Analytical Method** Consideration of the Magnetic field density as the product of the gradient with the magnetic potential vector $\vec{B} = \nabla \times \vec{A}$	**Semi-Analytical Method** Modeling of both magnets forces using amperian current model equations [23]	-	-	-
C.M.Saravia et al [9]	**Empirical/Numerical Method** Use of the FEM to model the magnetic field density and its mutual influence inside and outside all the magnets	**Empirical/Numerical Method** Real Measurements fitted into a power series $F_{mag}(x) = a_0 x + a_3 x^3 + a_5 x^5$	**Analytical Method** Use of Faraday's Law to obtain the induced voltage in the coil	**Numerical Method** Representation of the eletromechanical system in state space form and integration using the Runge-Kutta algorithm	**Analythical Method** Static consideration of viscous damping coeff. $F_a = c_a\dot{x}$, friction damping coeff. $F_f = c_p sign(\dot{x})$ and electromagnetic damping coeff. $c_e = \frac{(\sum \phi')^2}{R_{Total}}$

Table 2.1: LBEH modeling Summary I

Author[Article]	Magnetic Field	Interaction Magnetic Forces	Electromotive Induced Force	Coil Induced Current	Friction and Damping Forces
M.P. Santos et al [5]	**Semi-Analytical Method** Use of the equivalent surface current model and discretization of the magnet into a finite set of current loops. Use of elliptic integrals of first to third order	**Analytical Method** Derivation of the interaction energy between magnets recurring to the Bessel function to obtain the computation model	**Semi-Analytical Method** Consideration of the coil as a set of singular turns. Use of the Faraday-Maxwell equation to model the induced emf	-	**Analytical Method** Consideration and modulation of the magnetic damping force, and dry friction force using the Karnopp friction model
Kecik et al. [15]	-	**Numerical Method** Real Measurements fitted into a power series $F_{mag}(x) = \sum_{n=0}^{3} \alpha_n x^n$	-	-	**Analytical Method** Constant air flow damping coeff. c_a and dynamic electromagnetic couple coefficient $c_e(z(t))$
Wang et al [14]	**Semi-Analytical Method** Use of the equivalent surface current model and discretization of the magnet into a finite set of current loops. Use of elliptic integrals of first to third order	**Analytical Method** Derivation of the interaction energy between magnets recurring to the Bessel function to obtain the computation model	**Semi-Analytical Method** Consideration of the coil as a set of singular turns. Use of the Faraday-Maxwell equation to model the emf in the coil	-	**Analytical Method** Consideration and modulation of the magnetic damping force, and dry friction force using the Karnopp friction model

Table 2.2: LBEH modeling Summary II

2.3 ANN on dynamic system modeling

Artificial Neural Networks are a prominent Machine Learning algorithm, which allow for the computation of a quite extensive range of arithmetical and logical functions[10].

Machine learning algorithms are used nowadays in many application, such as function approximation, pattern recognition, prediction or clustering [24]. The use of ANN in dynamic system modeling falls under the Prediction category, for which Feed-Forward Multi-Layer ANN are most frequently used. Most reviewed studies use ANN dynamic systems models as a control implementation tool [11, 25, 26]. Furthermore this method can also be used for system identification analysis or behavior prediction[10, 27]. This applies to nonlinear dynamics such as mass-spring-damper or magnetic levitation systems. However, besides the wide application range of this methodology, its direct application on LBEH systems modeling has not been registered in the literature .

Chapter 3

Method

3.1 Energy Harvester Prototype

A magnetic levitation based harvester for low frequencies application is proposed in this section. The prototype developed in this work takes into consideration existent devices that use nonlinear properties from magnetic levitation to maximize motion potential, thus maximizing outputs [5, 14, 15, 18].

In this work, the presented architecture seeks to solve inherent questions to the energy harvesting viability issues. Reliability, low maintenance requirement and low intermittence, are relevant subjects on the approach taken during the development and construction of the device.

3.1.1 Dimensioning of the harvester

When looking to develop an energy harvester prototype with a certain architecture, dimensioning the device for intended applications is a step of major impact on its overall performance and viability [7, 8].

Theoretical LBEH intended application conditions and inherent properties where used as development foundations. The emulation of such inputs into an oversimplified model of the device helps to have a reference of its behavior for certain overall measurements, hence being a very useful tool for the attribution of dimensions to the prototype. For this work, a few *a priori* assumptions where used, in order to perform simulations and obtain an approximate response of the developing generator, for certain conditions.

Length dimensioning

The intended high inertial properties (when compared to other magnetic levitation devices [5, 28, 29]) required for the harvester, defined some construction and dimensional principles, which were used for initial size attribution in performed simulations. For the sake of compactness and prototype construction viability, a dimensional range was set from 200 mm to 300 mm. This gap was used for posterior simulation and result comparison.

Taking only into account the gravitational, centrifugal and magnetic interaction forces, an oversimplified model was sketched as can be seen in figure 3.1. The horizontal disposition of the device comes from its main goal application being for rotation purposes, from which its analysis in this configuration becomes important.

Considering this scheme, where a referential x is considered at the center of the device, a governing equation 3.1 was formulated. In this model, friction and damping forces where considered

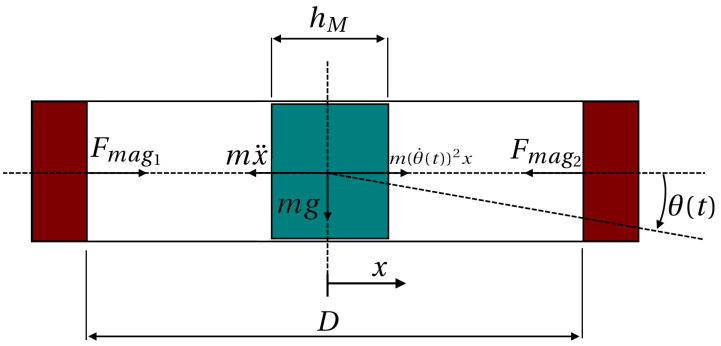


Figure 3.1: Simplified analysis to horizontal magnetic levitation system

negligible, in order to perform a simple initial analysis to the dynamics of the magnet [5, 9].

$$m\ddot{x} = F_{mag_1}(x) - F_{mag_2}(x) + m.g.\sin(\theta(t)) + m.(\dot{\theta}(t))^2.x \tag{3.1}$$

The simulation was performed using the software Simulink®, and from the formulated governing equation, the following block diagram was assembled (figure 3.2).

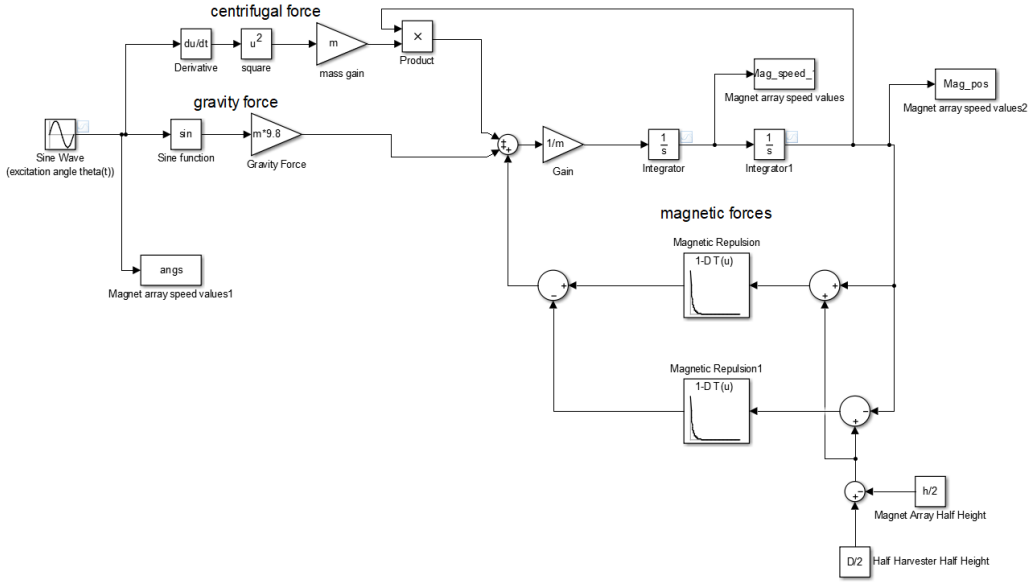


Figure 3.2: Horizontal levitation block diagram

For the input angle variation, values of ± 0.35 radians (± 20 degrees) and 0.5 Hz frequency were selected as average values for a desired sinusoidal external excitation. Its output multiplied by the gravitational force emulates the harmonic acceleration applied to the levitating magnet. Additionally, the centrifugal force resultant from rotation of the device is introduced to the system. Solving the governing equation 3.1 for the acceleration, $\ddot{x} = \frac{1}{m}(F_{mag_1} - F_{mag_2} + m.g.\sin(\beta) + m.\omega^2.x)$, the remaining influential forces to the acceleration of the magnet are obtained.

Since the fixed magnets influence (F_{mag_1} and F_{mag_2}) is function of the position of the levitation magnet, this variable must be properly manipulated and applied to a function that characterizes this force in relation with the distance to the magnets. The specific distance between the levitating magnet and both fixed magnets is calculated considering a referential in the center of the

harvester, (figure 3.1).

The force-displacement function of the end magnets was then introduced to this simulation using a Lookup table, into which, values from a magnetic force computing script were loaded [5]. These values were computed using properties from hypothetical applicable magnets, possible to see in figure A.1 and A.2 on appendix A. The plotted values for this function can be visualized in figure 3.3.

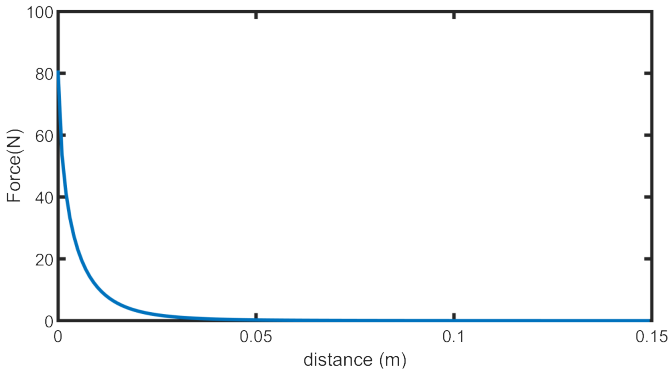


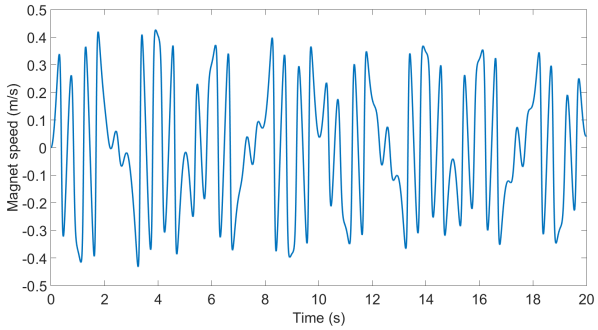
Figure 3.3: Magnetic repulsive force as a function of displacement

The fully programed block diagram, allowed the execution of simulations for a variety of lengths for the device, with constant values for the magnet of respectively 35 mm and 100 grams for height and mass.

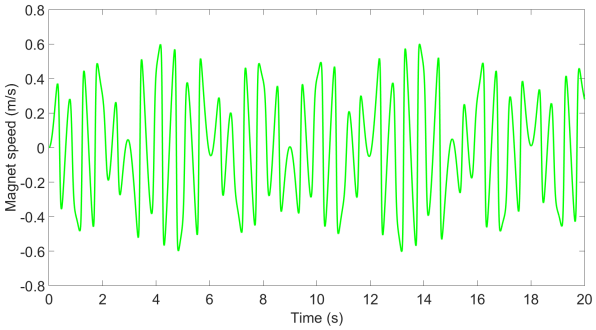
From those simulations, a few scope diagrams describing the speed of the magnet over time can be seen in figure 3.4. The evaluation of these results was made with view to achieve balance between performance and size.

From the displayed graphs it is possible to conclude the speed improvement from the length of 220 mm to 230 mm. While in the first configuration the levitating magnet reaches maximum speed values of approximately 0.4 m/s, the second one reaches magnitudes of almost 0.6 m/s, which represents an improvement of about 50%. The speed improvement for the other two lengths isn't so noticeable due to the similar maximum speed magnitude presented in both graphs. Therefore the 220 mm configuration was set aside, defining a minimum length of 230 mm for the device. The enlargement of the harvester does not enhance these values significantly, thus the maximum length of the device was set as 250 mm. As it is possible to realize from figure 3.5, the maximum speed value for the 250 mm configuration does not vary much from the 270 mm, hence the performance improvement ratio with size enlargement does not necessarily pay off. Also, increasing the variable range of the device would add extra complexity to its design and construction.

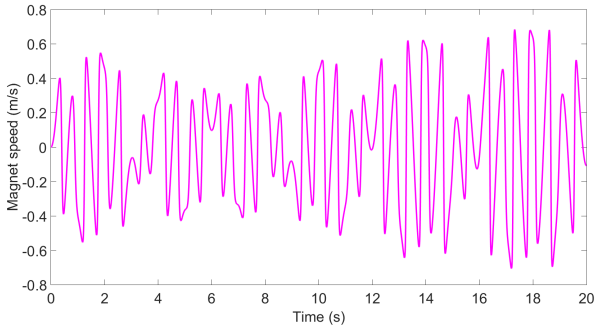
Apart from speed evaluation, the simulation also allows the visualization of the position response through time for the magnet inside the device. In figure 3.6, the position of the magnet over time related with the input angle sinusoid is shown for a configuration of 230mm, permitting a brief idea of its behaviour inside the harvester when stimulated with that specific excitation.



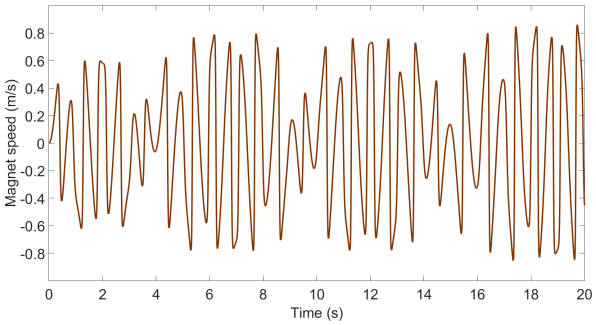
(a) Speed scope for a length of 220 mm



(b) Speed scope for a length of 230 mm



(c) Speed scope for a length of 240 mm



(d) Speed scope for a length of 250 mm

Figure 3.4: Speed simulation plots for various lengths of the harvester

Coil dimensioning

Optimizing components such as coil size is a complex analysis that needs dedicated work on the subject [30]. Therefore, a simplistic analysis was taken under consideration to dimension this component. The coil slots dimensioning was made with view to maximize the output potential of the generator. Maximizing coil turn number was one of the main goals in this topic. It was assumed that a configuration in which the coils' length is larger than the magnet array height hinders the generated voltage due to inverse magnet field variations on the coil, thus diminishing the induction effect [1]. Therefore, a coil with maximum loop number was considered, with the restriction that its length should not exceed the magnet array height.

These assumptions where introduced as further design requirements, enabling the dimensioning of the coil in relation to used magnet array size and other external variables.

3.1.2 Structural design

After the presented theoretical analysis, the dimensional features for the prototype were set. This step allowed the conception of the first sketches, permitting the search for applicable materials and parts for its construction.

Due to low frequency intended implementation, and rotation application purpose, some initial aspects were established in order to initiate the design of the device. The choice of electromagnetic induction as the transduction process, due to its functioning and constructional simplicity, allied with a magnetic levitation based architecture defined the starting sketch for the development of the harvester.

Three additional requirements where introduced in the developing concept, which affected

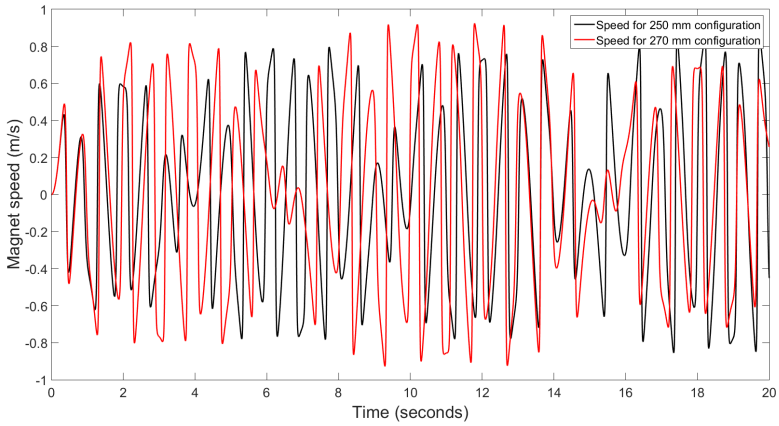


Figure 3.5: Magnet speed response comparison for 250 mm and 270 mm configuration

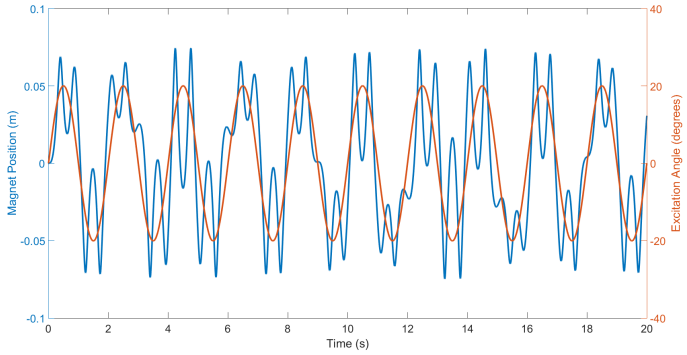


Figure 3.6: Position of the magnet over time

the design of the harvester. The idea of a guiding system for the levitation magnet was introduced based on other harvesters already developed [2, 12] and had the purpose of diminishing mechanical frictional losses from magnet and inner walls contact. Another desired functionality was the possibility to vary the length of the harvester for alternative configuration testing. The last design influential feature comes from the need of acquiring the position of the levitating magnet over time. Also, optimization assumptions like the minimization of radial distance between levitating magnets and coils, or the necessity of maximum smoothness on the inner surface must be taken into concern.

With the dimensioning of the device performed, and the overall functionality defined, a few material availability restrictions where introduced to the design plan, due to dimensional or technical constrains. The first defined parts were the fixed and levitating magnets. Ring type fixed magnets with diameter of 27 mm where chosen in order to allow a guiding shaft for the moving magnet. For the levitating magnet array, 25 mm magnets with internal threads were selected to enable the assemble between these magnets and the guiding shaft. Specifications of the magnet can be seen in figure A.1 and A.2, on appendix A.

From the defined topics and rules for the development of the harvester, a few sketches where made taking into consideration its overall dimensions and design properties. In figure 3.7 it is possible to visualize a first CAD sketch made for a possible device construction.

This first prototype sketch, was made to provide a concept for the proposed device. All required features where introduced to this model, however it contained several construction and

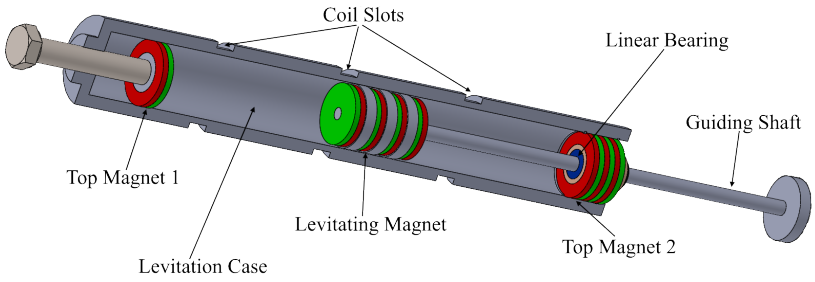


Figure 3.7: First CAD of possible harvester configuration

functionality limitations.

Although fixed magnet 2 is constrained with 2 elastic rings, a bolt assembled to fixed Magnet 1 enables its position regulation, adjusting the height of the levitation case. However, the length variation asymmetry would probably result in output diminishing by the coil close to the adjustable fixed magnet, since the magnetic flux through it would decrease. Another technical limitation of this model lies in the unavailability of parts with a 27 mm inner diameter needed for the assemble of the fixed magnets, and an exterior diameter suitable for the machining of relevant coil slots. Another critical functional flaw is the positioning of the linear bearing inside the fixed magnet 2, which would result in the magnetic interaction between magnet and metallic balls, increasing friction on the guiding shaft system.

Taking into account the number of variables present in the development of a device of this nature, and evaluating all the flaws detected in this concept model, it was necessary to schematize the variables that influenced the design of the harvester.

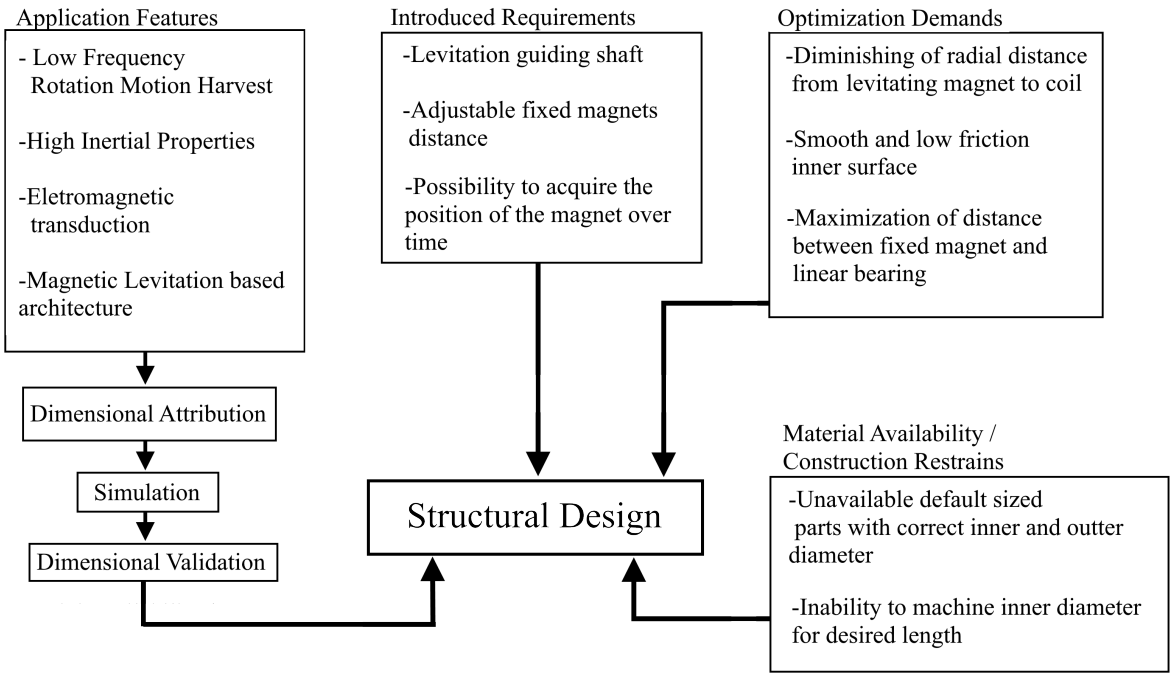


Figure 3.8: Device design scheme with associated variables

The analysis of figure 3.8 makes it easier to take into consideration all the internal and external features that conditioned the development of the device. The introduction of Material Availability and Construction Restrains parameters is of primary importance when considering the concep-

tion of a prototype, as it will define the material selection and will posteriorly influence the overall design.

From the referred limitations in terms of material and development possibilities, it was concluded that the inner surface body and the external coil support body would have to be assembled from different parts, one definitive construction feature, defined as a base for the new design. This decision was made due to the impossibility of machining the correct inner diameter for a piece with the required length, also, the possibility of machining it in two or three parts was dismissed, as it would be very hard to assure coaxiality between parts. With these new pre-sets defined, a new concept was projected, similar to the previous model but with the necessary improvements and alterations. After a few design enhancements and several sketches, a conceivable device was modelled. A section view of this new configuration can be seen in figure 3.9.

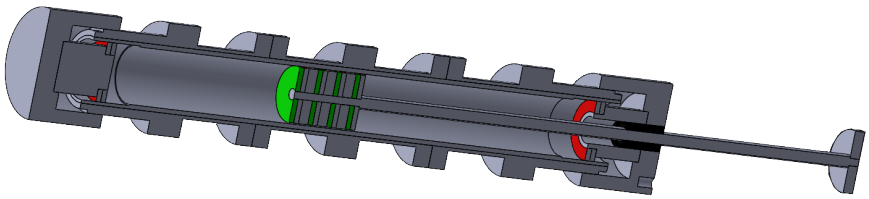


Figure 3.9: Final Energy harvester prototype CAD

This new configuration solves all the enumerated issues in the previous model analysis. Not only the construction inconsistencies were solved due to previous material selection, but also other situations related with the linear bearing assembly and the displacement variation system of the end magnets. However, from this new configuration, some disadvantages arise. One of the most relevant topic is the unavoidable enlargement of the distance between magnet and coil, due to the fact of the body being composed of an internal (acrylic tube) and external part.

Regarding the presented solutions, the displacement variation system for the end magnets was solved creating two internal threaded caps in which the end magnets assemble. The thread on the caps allows for the coupling of those components with the main body, simultaneously allowing the length regulation of the device. The assembly between magnets and caps was made using two cylindrical parts that would attach to both magnets and caps making use of a strong adhesive. As it is possible to visualize, one of the caps as well as the corresponding cylindrical part, are drilled in order to allow the assembly of the bearing for the guiding shaft. Furthermore, new coil dimensions where assigned to the prototype, assuming a 5 magnet array with an approximate height of 35 mm, it was possible to maximize its dimensions using the coil dimensioning assumptions presented earlier in section 3.1.1.

In order to acquire the position of the levitating magnet, an external structure was assembled to the drilled cap. This structure, as can be seen in figure 3.10, has the purpose of holding a sensor unit, which measures its distance to the disc inserted in the tip of the shaft, allowing to track the position of the levitating magnet.

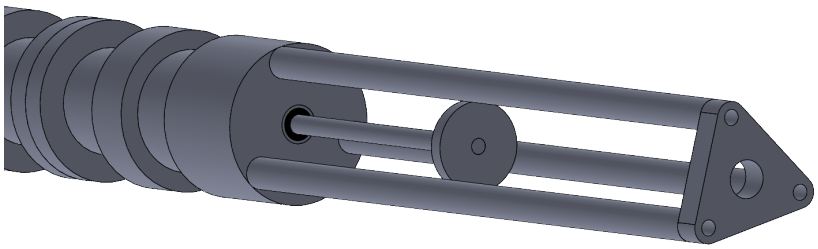


Figure 3.10: Sensor support structure

This structure ensures a steady position of the sensor in relation to the harvester, permitting the acquisition of accurate data for any desired external excitation.

3.1.3 Construction of the device

As referred in the previous section, the development of the device was made taking into consideration some material availability issues, figure 3.8. The need for a low friction inner surface, and the inability to machine it without the required precision, lead to the mandatory choice of a component with correct inner diameter and surface smoothness for the previously picked magnets. Since the diameter of the levitating magnet is 25mm, an acrylic tube with dimensions of 26 x 30 mm in diameter was chosen to perform the levitation case.

Regarding the magnets, the already referred 27 x 16 x 5 rings were picked for the fixed magnets, and 25 x M5 x 5 threaded pot magnets where chosen to couple for the levitating array configuration.

The material for the three parts that compose the external body had three main requisites: it could not be ferromagnetic (preferentially non-metallic) in order to cause no interference in the motion of the magnet; it should present low degradation to elements exposure; and be easy to machine. For these parts, the engineering plastic Ertalon was selected for the referred properties as well as dimensional availability. From a tube with diameter dimensions of 50x30, three parts where machined. The same procedure was performed for the remaining plastic components (caps and respective cylinders, shaft tip disc and sensor coupling part), yet, a nylon rod was used instead as raw material.

For the remaining components like the levitation guiding shaft and sensor support structure, steel and aluminum rods where used respectively. The necessary threads were then machined in order to allow the assemble of the different parts.

The coils were custom made for this generator using AWG 35 wire (0.1426 mm of diameter). The wire selection was made with loop number maximization in view, but considering a larger wire diameter than the one considered by Soares dos Santos et al. [5] to avoid large coil internal resistance. The only non processed component was the SKF linear bearing, chosen in order to fulfill the dimensional need imposed by the diameter of the levitation shaft. Figure 3.11 presents the developed generator after full assembly.

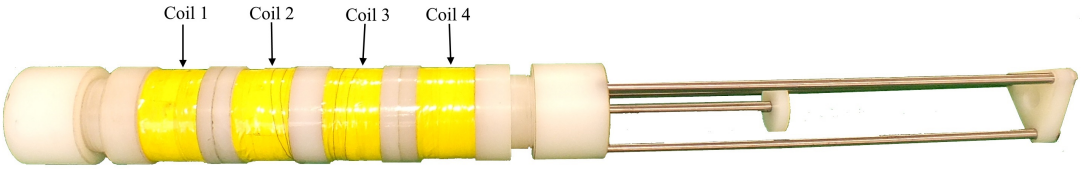


Figure 3.11: Assembled Generator prototype

The different drawings for the components are presented in appendix A.

3.2 Testing Station

In order to draw useful information from the system, it is required to simulate real application conditions. The main objective for this testing station is to record the position of the magnet over time inside the generator, for a correspondent stimulation angle. For this purpose, a structure with respective actuator and sensing module was developed. This section explains the development of the testing structure, as well as its programming and the data acquisition process itself.

3.2.1 Development and Construction

Actuation Module

The actuation module, as can be seen in figure 3.12, composed by an aluminum base, a servo motor, a metallic arm and an acrylic plastic grip, has the objective of applying sinusoidal stimulations on the generator for a desired Amplitude and Frequency.

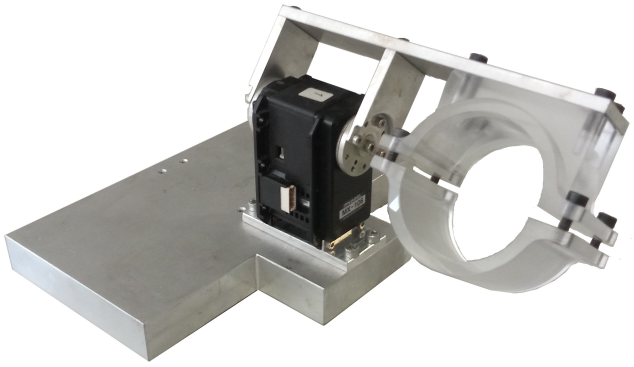


Figure 3.12: Testing Station Structure

The referred servo-motor is a Dynamixel® MX106-R, which has an embedded closed loop PID control system for a wide range of intrinsic variables such as position, speed among others [31]. This device has a maximum torque of 80 kg.cm, which is enough for the application in this work. However, if the need for testing a bigger model of a generator arises, the base has a set of 4 extra holes in which it is possible to assemble a second motor with just a few component alterations.

Sensing Module

Acquiring the position of the magnet over time required a device able to measure the distance to an object with reasonable precision and sampling rate.

A microsonic® Nano 15/CU ultrasonic sensor was picked considering its adequate technical features and size.

Figure 3.13: Microsonic Nano 15/CU

This sensing device has a 20mm to 250mm measuring range, compact size, good sampling rate and precision, making it suitable for the intended application. The output is a 0-10 linear voltage which can be regulated for different intervals using a teach-in pendent, however the configuration and data acquisition method will be discussed later in this work. See figures A.4 and A.5 in appendix A for more detailed informations on the sensing unit.

Control and Data Acquisition Unit

In order to control the servo motor and record the data acquired by the ultrasonic sensor, a Control Unit was developed using an Arduino® NANO board. The electric scheme of this unit can be seen in figure 3.14.

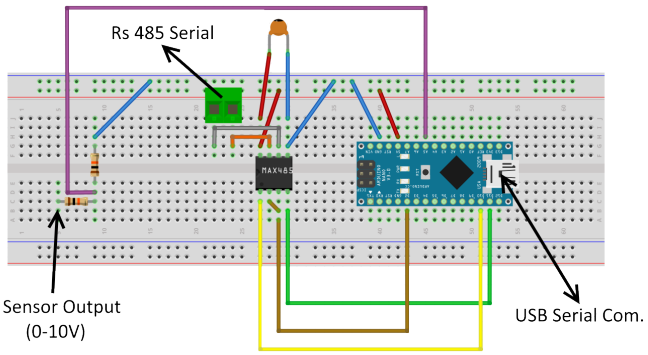


Figure 3.14: Testing Station Electric Circuit

The Control board (Arduino®) performs the following operations:

- Calculation of the external angle excitation to apply over time
- Communicates calculated angle values with the Servo motor, via Rs485

- Reads analog inputs from ultrasonic sensor
- Sends both angle and position information to laptop via serial communication

The Rs485 communication integrated circuit, MAX 485, as can be seen in figure 3.15, has the objective to differentiate the Serial communication signal, as is required by the RS485 electrical standards.

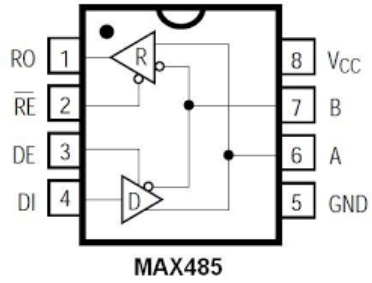


Figure 3.15: Max 485 Integrated Circuit Scheme

This electronic component is powered on pins 8 and 5 (Vcc and GND) from the 5V Source provided by the Arduino®. The RE and DE pins control the direction of information flow (Tx or Rx), while the pins DI and RO are the physical connections for sent or received information respectively. Pins A and B are the differential signal receivers or emitters, according to the communication type established (Tx or Rx). The use of a capacitor parallel with the power source is to keep a steady voltage input, in order to avoid noise during the communication process.

The Resistive Voltage Divider has the purpose of reducing the Sensor Output range from 0-10V to 0-5V in order to allow analog readings from the Arduino®.

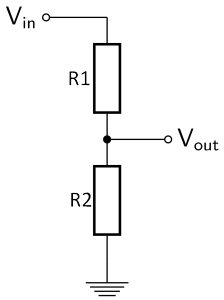


Figure 3.16: Resistive Voltage Divider

As it is possible to see from figure 3.16, this circuit makes use of two Resistors, where R1=R2 in order to ensure and equal voltage drop in both resistors, confining the potential V_{out} between them in a 0-5V interval.

3.2.2 Programming and data Recording

In this section the main Control Unit program will be briefly explained. Also, the configuration of the sensor and output reading will be talked through. The data collecting and organizing methods will also be taken into consideration in this section, for they are an important part of the data preparation process for neural network training.

The Station main program was made recurring to Arduino® IDE, therefore the script was wrote, debugged and uploaded from this platform. It was programmed to perform all the 4 referred operations once per cycle. This is important in order to synchronize data computed by the Arduino® and data generated by the Sensor.

Angle calculation

In order to reply real application stimulations, sinusoidal low frequency rotations must be applied to the device. For this effect, a time dependent sine wave formula was implemented in the Script based on the following equation 3.2.

$$\theta(t) = a\sin\left(2\pi f t + \varphi\right) \tag{3.2}$$

This equation describes the angle over time, for a wave with desired features.

The wave emulation strategy consists in sending these computed wave angles to the servo motor every cycle. Since the cycle time of the code is approximately 10 ms, theoretically the Servo will move to a new angle approximately 100 times per second with constant speed, performing an approximation of the stimulation the generator would get in a real application.

For training purposes, there is also the need of applying different waves in a synchronous way, as will be explained further in section 3.2.2. This means that a continuous wave with variable parameters over time must be sent to the servo. Non continuous wave excitation would result on invalid system inputs on real applications. Hence, varying Period and Amplitude on the excitation wave without abrupt angle changes is mandatory. The script excerpt presented in figure 3.17 contains the code for four waves with different periods and same amplitude variation over time.

```
while(millis()<waitingtime){} //Waiting routine
tempo = millis() / 1000.0; // Time calculation (seconds)
while(tempo>SimulationTime+waitingtime_s){} //Simulation stop routine

if(tempo<TimePeriod+waitingtime_s){
tempow1=tempo-waitingtime_s; //Time set as 0 for wave 1
maxwaveangle=int(tempow1/10)*5+15;
waveangle = maxwaveangle * sin((6.28 / Periodw1) * tempow1);
}
else if (tempo>TimePeriod+waitingtime_s && tempo<2*TimePeriod+waitingtime_s) {
  tempow2=tempo-TimePeriod-waitingtime_s; //Time set as 0 for wave 2
  maxwaveangle=int(tempow2/10)*5+15; //Wave amplitude variable over time (+5 every 10 seconds)
  waveangle = maxwaveangle * sin((6.28 / Periodw2) * tempow2); //Second wave computation (Periodw1)
  }
else if (tempo>2*TimePeriod+waitingtime_s && tempo<3*TimePeriod+waitingtime_s){
  tempow3=tempo-2*TimePeriod-waitingtime_s; //Time set as 0 for wave 3
  maxwaveangle=int(tempow3/10)*5+15; //Wave amplitude variable over time (+5 every 10 seconds)
  waveangle = maxwaveangle * sin((6.28 / Periodw3) * tempow3); //Third wave computation (Periodw1)
  }
else {
  tempow4=tempo-3*TimePeriod-waitingtime_s; //Time set as 0 for wave 4
  maxwaveangle=int(tempow4/10)*5+15; //Wave amplitude variable over time (+5 every 10 seconds)
  waveangle = maxwaveangle * sin((6.28 / Periodw4) * tempow4); //Fourth wave computation (Periodw1)
  }
```

Figure 3.17: Variable parameters wave computing

The four different waves come from the need of synchronizing the waves when varying the Period. A TimePeriod is defined in order to establish the time each wave lasts. When that time ends, the Period is changed and the wave time is zeroed to guarantee that all waves start at angle 0. Wave periods were also selected with synchronization in view.

The amplitude is varied over time, with a function that sets the starting amplitude interval as (-15,15 degrees), and increases this value by 5 degrees every 10 seconds. The more a wave lasts, the greater the amplitude value it reaches before restarting with a new period.

Communication with Servo

In this section, the communication with the used servo motor is discussed. The Dynamixel MX 106-R has a 4 pin connector, as can be seen in figure 3.18, in which two pins (Vcc and GND) Power the motor with 24 V, and the other two (D+ and D-) perform the RS 485 communication interface.

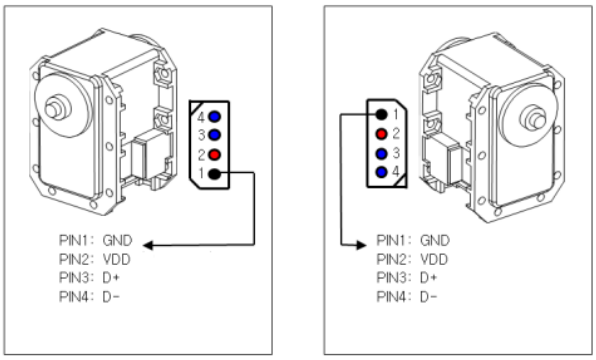


Figure 3.18: MX-106 Connector

The communication with the device is made via data packages. In order to allow communication with the motor and computer, a SoftwareSerial object Rs485Serial was declared. The circuit presented in the previous section 3.14, takes care of the signal differentiation, therefore the information send from the Arduino is performed using a simple Rs485Serial.write command, which sends Serial data with the defined baud rate. The data package configuration can be seen in figure 3.19.

0xFF	0xFF	ID	INSTRUCTION	PARAMETER 1	...	PARAMETER N	CHECK SUM

Figure 3.19: MX-106 Instruction package [31]

This package consists in 2 sets of 8 bits set to 1 (2 sets of FF in hexadecimal), which trace the message start. The next bytes contains the ID of the motor, the length of the message ($Length = number\ of\ parameters + 2$), and the instruction desired to perform, respectively (see figure 3.20).

For this specific application, the used instruction is strictly the WRITE_DATA, since the only exchanged information was the desired positions sent to the motor.

The first parameter to send corresponds to the memory address of the variable which is desired to change. In this particular case, the Goal Position addresses are the 0x1E and 0x1F, however there is only the need to send the first register in the package. The last byte in the instruction data package (CHECKSUM) is used to check if the information sent during communication is not damaged. The logical operation to construct this byte is described in the following equation, $Check\ Sum = \sim (ID + Length + Instruction + Parameter1 + ... + ParameterN)$, where "~" is the Not Bit operator. Taking this into consideration, a function was developed in order to automatically compute the check sum byte for a given data package, it can be visualized in figure A.6 on annexed documents.

Value	Name	Function	No.of Parameters
0x01	PING	No execution. It is used when controller is ready to receive Status Packet	0
0x02	READ_DATA	This command reads data from Dynamixel	2
0x03	WRITE_DATA	This command writes data to Dynamixel	2 or more
0x04	REG WRITE	It is similar to WRITE_DATA, but it remains in the standby state without being executed until the ACTION command arrives.	2 or more
0x05	ACTION	This command initiates motions registered with REG WRITE	0
0x06	RESET	This command restores the state of Dynamixel to the factory default setting.	0
0x83	SYNC WRITE	This command is used to control several Dynamixels simultaneously at a time.	4 or more

Figure 3.20: MX-106 Possible instructions [31]

Since the motor has 12 bits in the goal position register (4095 positions in 360 degrees), the output angle from the sinusoidal function (3.2) must also be processed in order to be sent to the motor. The first step is to convert the sinusoidal angle output into viable angles for the motor. Considering the zero position of the servo as 180 degrees, the value computed from the sinusoidal function must be add to this value, $desired\ angle = 180 + \theta(t)$. After this step, it is necessary to send the $desired\ angle$ value in a correct manner to the registers of the motor. This is done using another created function that converts the resultant float value into proper hexadecimal packages. The function created for this effect can be seen in the figure A.7, presented in the annexed documents.

With all the data processing performed, the packages are sent byte by byte using the Rs485Serial Communication object inside a *for* cycle.

Sensor Calibration

The sensor calibration is described in this section, it defines the displacement interval in which the sensor will output the entire analog range. The transformation of the displacement measured by the sensor into correct levitating magnet position, is also talked through.

In this work the whole sensor range was needed, consequently the sensor was configured to measure from 20 to 250 mm. Figure 3.21 shows the steps followed in order to configure the sensor unit. The left column steps were followed in order to set the output scale to vary from 0 to 10 V in the intended interval. However, for Arduino® reading purposes, this output had to be restrained between 0 and 5 V, for which the previously referred Resistive Divider was applied, figure 3.16.

In order to obtain distance measurements from the sensor, the resistive divider output was injected into an analog input pin. The Arduino® nano board uses a 10 bit adc converter, from which will result a reading of 0 to 1023. The measurement range of the sensor was related with the reading performed by the Arduino®, resulting in the linear graph, presented in figure 3.22.

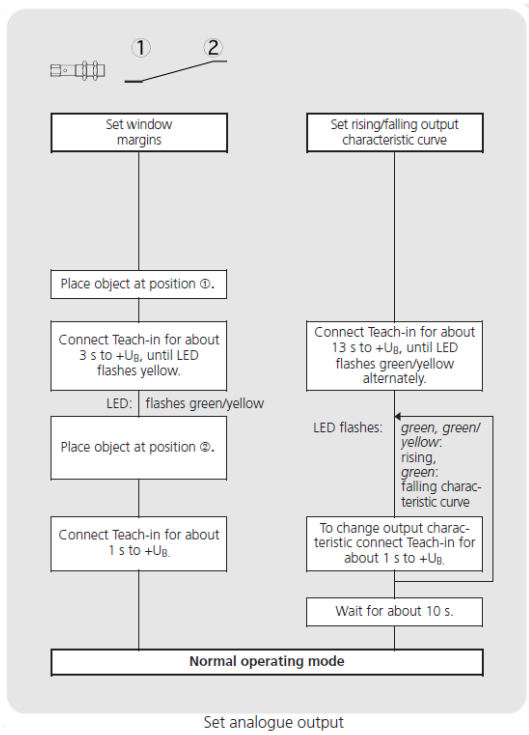


Figure 3.21: Steps to configure the measuring range of the sensor unit

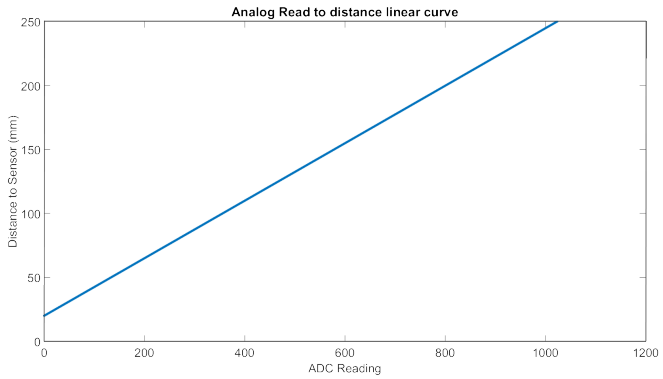


Figure 3.22: Analog Read over Sensor distance curve

From the displayed line, the equation for the ADC reading to distance relation can be formulated (3.3), from which it is possible to accurately calculate the disk distance to the Sensor, from now on denoted as x_{Disk}.

$$x_{Disk} = 0.2248 * ADCread + 20 \tag{3.3}$$

However, this value would not correspond yet to the real position of the magnet inside the generator, therefore, a geometrical transformation was applied to this values in order to calculate them relatively to the initially considered referential 3.1.

In the presented figure 3.23, it is possible to view the referential systems and measures considered for the referential exchange. The nomenclature of the values can be seen in table 3.1.

The goal of this step is to have a coordinate system in the center point of the generator, for

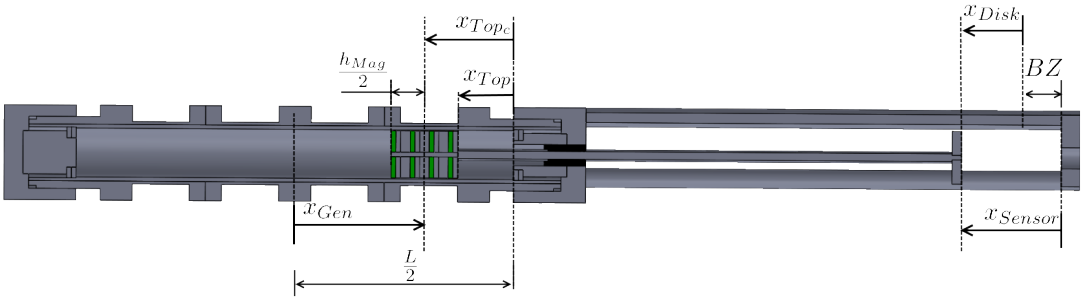


Figure 3.23: Analog Read over Sensor distance curve

Variables	Description	Value (mm)
x_{Sensor}	Disk distance to Sensor	-
x_{Disk}	Position of the disk relatively to Blind Zone mark	-
x_{Top}	Distance of levitating magnet to fixed magnet	-
x_{Top_c}	Distance of levitating magnet center to fixed magnet	-
x_{Gen}	Distance of levitating magnet center to the center of the generator	-
$\frac{L}{2}$	Half the inner length of the generator	115 to 125
BZ	Blind zone of the sensor	20
$\frac{h_{Mag}}{2}$	Magnet array half height	17.5

Table 3.1: Figure 3.23 variable description

agreement with previous analysis. This procedure was performed according to the following reasoning:

-Considering a x_{Disk} referential, which has origin in the blind zone line, and a x_{Sensor} referential which corresponds to the real distance from sensor to disk, the following applies:

$$x_{Disk} = x_{Sensor} - BZ. \tag{3.4}$$

-Considering a second referential system x_{Top}, now measuring the distance from the inner top of the generator and the top surface of the levitating magnet, it is also possible to assume that equation 3.5 is true.

$$x_{Disk} = x_{Top}. \tag{3.5}$$

This assumption is done considering equal displacement between referential systems and the points they are measuring, thus mutually canceling their effect on the referential transformation.

-Performing the measuring point exchange to the middle point of the magnet,

$$x_{Top_c} = x_{Top} + \frac{h_{Mag}}{2} \tag{3.6}$$

allows for a real measure of the center of the magnet to the right top of the generator.

-The last step consists in transforming the x_{Top_c} coordinates into values in the x_{Gen} referential system. For this, the following equation, 3.7 was applied,

$$x_{Gen} = -\left(x_{Top_c} - \frac{L}{2}\right) \tag{3.7}$$

which relatively to the true distance measured by the sensor is as follows, 3.8

$$x_{Gen} = -x_{Sensor} + BZ - \frac{h_{Mag}}{2} + \frac{L}{2} \tag{3.8}$$

Data Storing

All data from each simulation, both collected and generated from the Arduino® platform, was stored in .txt files using the open source software CoolTerm. This program reaches to the selected serial port and has the option of printing the received information into a desired format file.

The .txt file format was selected due to the simplicity of reading its content using MATLAB®.

Data Band Selection

Having the data collection process correctly implemented, it was now possible to generate and acquire the necessary data for the network training process. The input data range selection was made with view to cover most possible working amplitudes and frequencies. This line of thought was taken in order to provide the network the maximum application conditions for training.

Input wave was generated by the Arduino® while being applied as external excitation for the generator, the generated waveform can be seen in figure 3.24.

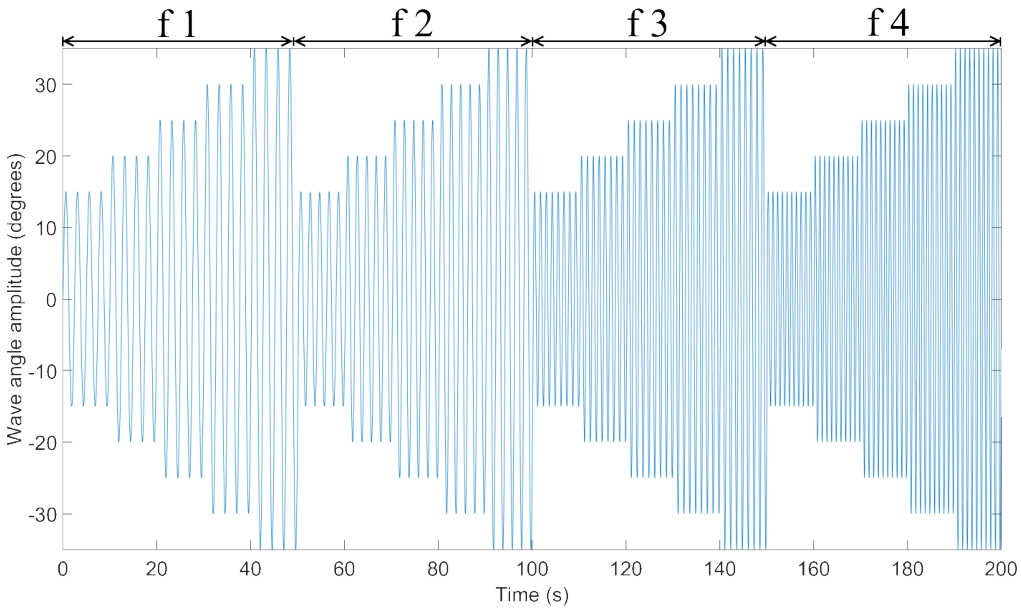


Figure 3.24: Input excitation wave

This generated sinusoidal input wave repeats the same angle amplitude sequence four times for four different frequencies. The angle and frequency variation can be seen in table 3.2, where the frequency values, were selected taking into consideration intended application properties and the synchronization between excitation waves, for continuity purposes. The time interval for amplitude change was set as 10 seconds for 2 main reasons. The first reason was synchronization (10 seconds correspond to a complete number of cycles for every selected wave), the second reason was to guarantee stationary regime in all waves. This means that, when a new excitation wave starts, the position of the magnet affects the behavior of the device, for that particular wave, in the first wave cycles. However it has been verified experimentally that 10 seconds were enough for the motion response to tend to a repetitive pattern of values.

Frequency (Hz)	Angle Amplitude (degrees)	time interval (s)
f 1=0.4	15	0-10
	20	10-20
	25	20-30
	30	30-40
	35	40-50
f 2=0.5	15	50-60
	20	60-70
	25	70-80
	30	80-90
	35	90-100
f 3=0.8	15	100-110
	20	110-120
	25	120-130
	30	130-140
	35	140-150
f 4=1	15	150-160
	20	160-170
	25	170-180
	30	180-190
	35	190-200

Table 3.2: Input excitation wave Properties over time

This phenomenon can be verified in figure 3.25, where the magnet presents a non uniform behavior for the first 2 cycles. It takes more time to stabilize in higher frequency bands, where the magnet does not reach the extremities of the generator, and is more accentuated in frequency transitions, where there is an abrupt negative amplitude variation.

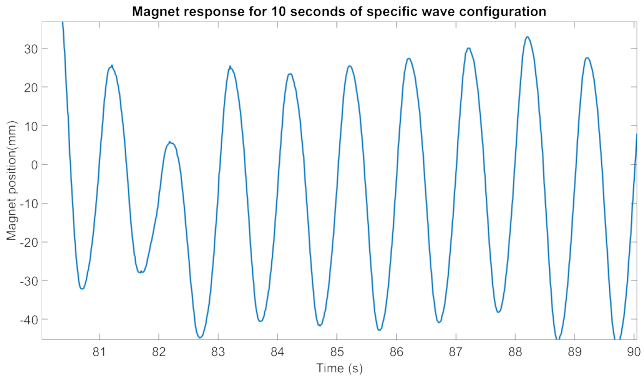


Figure 3.25: Magnet behavior for a particular 10 second period of a wave

From the selected input data, the acquisition of the position of the magnet over time provided the target data for network training. A representative plot of the behavior of the magnet over time for the given excitation wave is shown in figure 3.26.

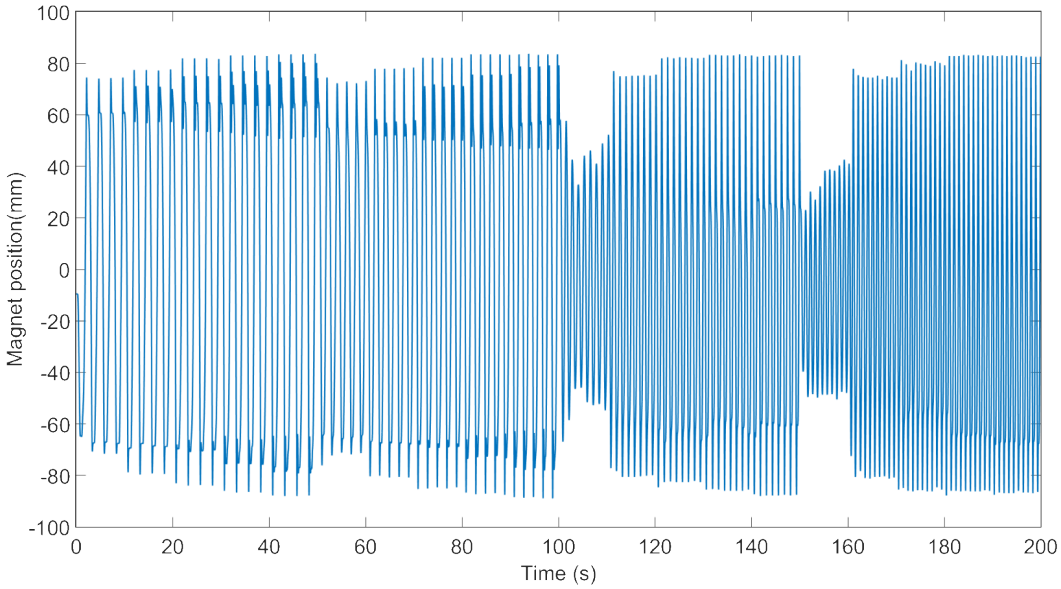


Figure 3.26: Magnet response to input excitation shown in figure 3.24

For an easier visualization of the motion response by the magnet, figure 3.27 shows the subplots for the four different frequencies contained in the input excitation wave.

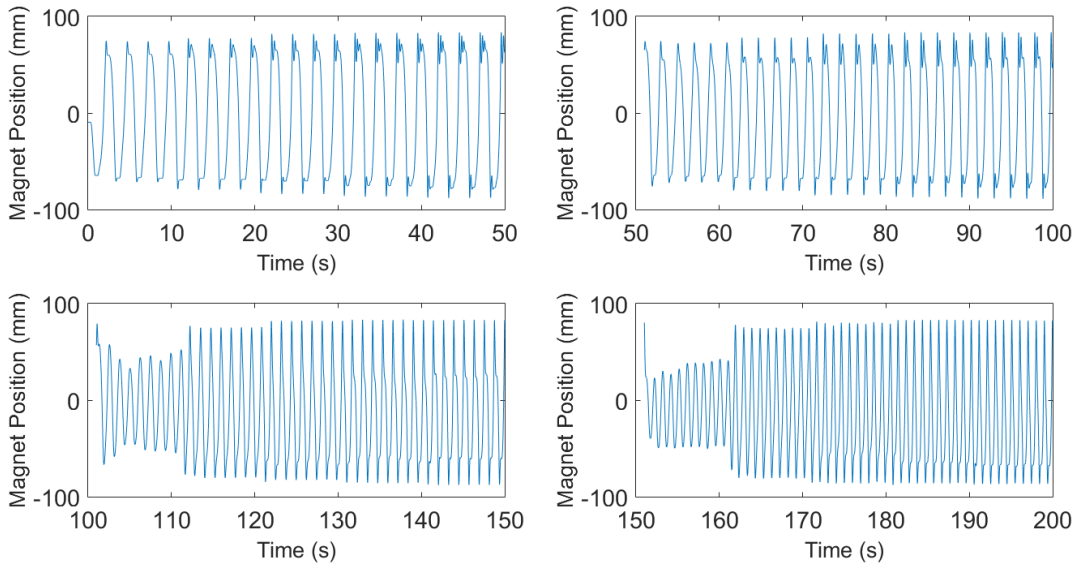


Figure 3.27: Magnet response to input stimulation of figure 3.24 (separated by frequency)

3.3 Dynamic system modeling using Artificial Neural Network

The Neural Network design and training algorithms where implemented recurring to the MATLAB software Neural Network Toolbox.

3.3.1 Network Architecture

The presented modeling problem falls under the category of time series analysis. While many network architectures are able to perform predictions for systems such as the one presented in this work, the NARX (Non-linear AutoRegressive model with eXogenous input) architecture is a popular approach for time series system modeling, mainly due to its good predictive and adaptive features [25]. Considering the available Network types provided in the used toolbox, the NARX architecture can be considered the most suitable strategy to approach this problem from a machine learning solution perspective. Another strong feature of this type of network is the fact that it can be trained using static backpropagation algorithms, since tapped delay lines at the input of the network can be replaced with an extended vector of delayed values of input and targets [24].

The following figure 3.28, represents a schematic view of this Network Configuration (in Closed Loop form).

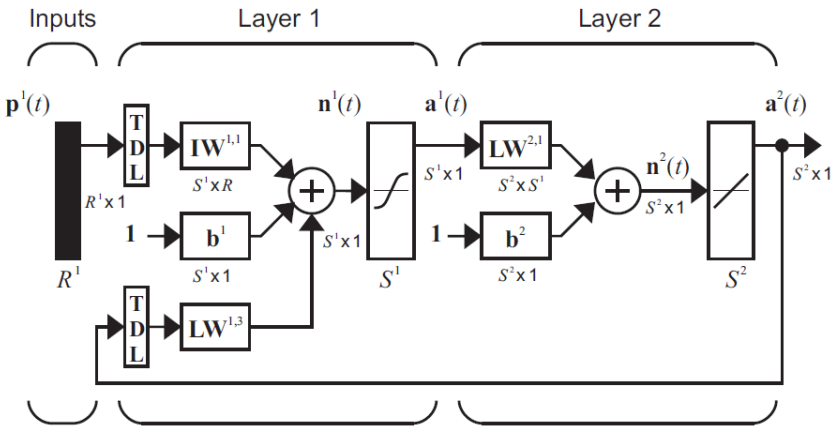


Figure 3.28: NARX general Architecture diagram [24]

Thedefining equation for this network model 3.9 [32], considering the input and target variables of this work as θ and x respectively, can be expressed as follows

$$x(t) = f(x(t-1),...,x(t-d_x),\theta(t-1),...,\theta(t-d_\theta)) \tag{3.9}$$

where d_x and d_θ correspond to the taped delay line maximum values for each input. In this particular application, the $\theta(t)$ and $x(t)$ correspond to excitation angle and position of the magnet respectively (inputs and targets).

This network architecture is composed by two layers. Layer 1 has 2 inputs, the input vector values and the target value backpropagation input. These inputs are inserted into Tapped Delay Lines (TDL), which will contain the variables $(\theta(t-1)),...,\theta(t-d)$ and $(x(t-1)),...,x(t-d)$ respectively. This layer also contains the hidden neurons with the correspondent connection weights and biases. The presented architecture considers a default tan-sigmoid transfer function as threshold for the output of this layer. Layer 2 has as input the first layer output, the same neuron number as Layer 1, correspondent weights and biases vectors, and a linear transfer output. Regarding the configurable parameters of the Network, it is possible to define the Hidden layer S^1 neuron number and the length of the tapped delay lines.

The following table 3.3, contains the NARX variable nomenclature.

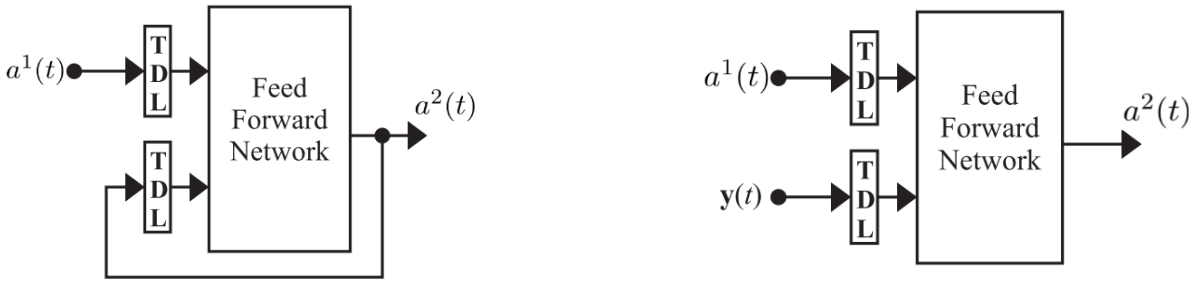
Considering this architecture as the most adequate available network structure for the imposed problem, the following steps consisted in defining the variable parameters (Hidden layer

Variables	Description
$p^1(t)$	Input Vector index 1 value
$a^1(t)$	First layer Output vector
$a^2(t)$	Neural Network Output value
S^1	Number of Hidden Layer Neurons
$y(t)$	Targets
TDL	Tapped delay lines
W	Weight matrix
b	Bias vector

Table 3.3: NARX architecture variables

neurons and TDL), defining the architecture, preprocessing the acquired data and selecting the most appropriate training algorithm.

Regarding the NARX model, the training process can be performed using two different configurations. The Parallel Architecture 3.29a, where the output of the network during training is fed back to the feedforward network, and the Series-Parallel Architecture 3.29b, in which the true output target is used instead of feeding back the estimated output.



(a) Parallel Architecture (Closed-Loop)

(b) Series-Parallel Architecture (Open-Loop)

Figure 3.29: Possible training an simulation NARX Forms [24]

Since the true output is available during training, the Series-Parallel form can be implemented. This architecture has two main advantages. The first is that the input to the feedforward network will be more accurate. The second is that the resulting network has a purely feedforward architecture, and static backpropagation can be used for training [11].

Taking into consideration such aspects, the network structure concept was mostly defined at this point.

3.3.2 Network Parameterization

In order to properly configure the parameters of a network, it is a good practice to first define the training method to use. For feedforward network structure there are three available training algorithms.

-Levenberg-Marquardt

-Bayesian Regularization

-Scaled Conjugate Gradient

In this work, the Bayesian regularization and the Levenberg-Marquardt algorithms were used to train different NARX networks. Prediction problems, such as the one in this work, are similar

to function approximation problems, which for the Bayesian Regularization algorithm is appropriate, considering multi layer network training in this kind of applications [33]. This algorithm also returns the effective number of parameters, correspondent to the number of total parameters (weights and biases) that are used during the training of the network, which is good to estimate the quality of the implemented structure for the presented problem.

The Bayesian Regularization algorithm also takes place within the Levenberg-Marquardt training process, hence this method was also used for network training, in order to conclude which learning algorithm is more appropriate for the presented problem. Although this last consumes more computation resources, its algorithm is the fastest among backpropagation methods in the MATLAB toolbox.

The selection of variable structural features for the network such as Neuron Number or tapped delay lines, must be done with the goal to maximize its performance, thus minimizing the mean square error of the output when compared to target values. Due to the relation between input data array size and possible parameter number(10000 or more data points to a maximum of 250 parameters), the chances of overfitting are very small, removing the overfitting possibility as a design constrain for the network [24].

As previously referred, the Neural Network and its associated training algorithms were implemented using the MATLAB Neural Network Toolbox. The developed code can be seen in appendix B.

The script presented in B.1, performs all the necessary steps for the training procedure. Operation sequence is performed according to the following points:

- Reads the acquired data files and converts the data into proper training input format
- Implements the desired parametrization (TDL and Hidden Layer Neuron number)
- Applies the data preparation steps, defines the data division method and applies the indicated ratios for training, validation and testing
- Enables the choice of the learning algorithm, epoch number, and also initiates the training process.
- Computes the error between targets and outputs, as well as the performance of the Network.

Table 3.4 presents the performance (mean square error), equation 3.10, of several training parameters for the two referred training algorithms, different initial conditions (Random initial Weights and Biases values) and different variable parameters. Since Bayesian Regularization does not have Validation, the Epoch limit has been set as 1500. It has been verified experimentally that increasing those values does not improve the training process and the chances of overfitting are increased.

$$mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - a_i^2)^2 \tag{3.10}$$

This table has the purpose to aid in the analysis of training performance between networks with different parameters and learning algorithms. Its analysis also helps to correctly define the most suited parametrization configuration.

Analyzing the table it is possible to conclude that the Neuron number increase does not necessarily represent an error decrease for the learning process. However, the enlargement of the TDL delay, results in good performance enhancement (decrease).

A further analysis to the effect of TDL increasing in the network performance, suggests that this enhancement tends to become irrelevant when applying more than 7 delay lines, at which value is possible to obtain performances around 0.29 for both training algorithms.

		Levenberg-Marquardt				Bayesian Regularization			
HLN	TDL	performance values			mean perf	performance values			mean perf
10	2	1.1984	1.0848	1.2589	1.1807	1.3430	1.2971	1.2905	1.3102
	3	0.6502	0.8653	0.8203	0.7786	0.8500	0.6452	0.6339	0.7097
	4	0.5670	0.5842	0.5479	0.5663	0.5792	0.5660	0.5506	0.5653
	5	0.3579	0.3470	0.3879	0.3643	0.3430	0.3486	0.3355	0.3424
11	2	1.0774	1.0536	1.0129	1.0480	1.3633	1.2685	1.4323	1.3547
	3	0.6279	0.6588	0.6403	0.6424	0.6086	0.6119	0.6262	0.6156
	4	0.4981	0.6106	0.4606	0.5231	0.5395	0.5630	0.5455	0.5493
	5	0.3575	0.3522	0.3867	0.3655	0.3271	0.3328	0.3250	0.3283
12	2	1.2431	1.1210	0.9150	1.0930	1.2806	1.3482	1.3296	1.3195
	3	0.6317	0.6324	0.6840	0.6494	0.6183	0.6082	0.6060	0.6108
	4	0.5340	0.4927	0.5977	0.5415	0.5414	0.5378	0.5488	0.5427
	5	0.3496	0.3536	0.4931	0.3988	0.3283	0.3139	0.3398	0.3273

Table 3.4: NARX learning performance for variable parameters

Considering the computation and training time an influential development quality feature for a Network, it must be projected to use the least computational resources possible. Therefore, the selection of a 10 Hidden layer Neurons structure and the implementation of 7 Tapped delay lines seemed to be the most suitable configuration for the presented problem.

Training Validation

The training process provides various validation tools that when properly studied provide important performance indicators of the implemented network quality. In this training validation, the following tools where analyzed:

- Regression- Relation between Training, Testing and Validation data Outputs with Targets.
- Error Autocorrelation- Correlation between prediction error and previous time-steps errors
- Cross-correlation- Correlation between prediction error and input sequence

The following analysis compares these training outputs for both used learning algorithms. In order for the network to be properly validated, the following properties must be confirmed. Regression values must be as close as possible to 1. The prediction errors, $e(t) = y(t) - \hat{y}(t) = y(t) - a^2(t)$ must be uncorrelated with each other from one time step to another, and the prediction errors must also be uncorrelated with the input sequence [34].

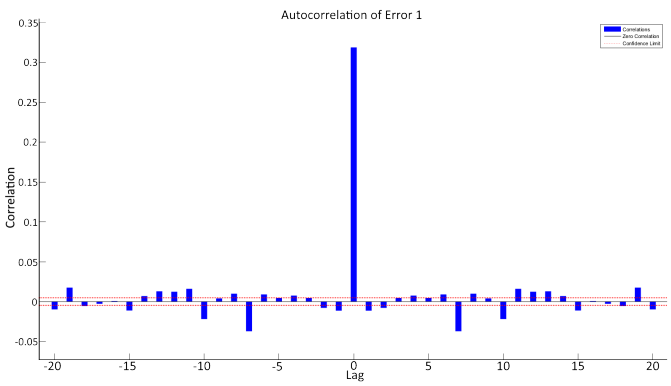
Regression provides a measure of training and testing data fitting to the targets. The following table, 3.5, shows average values for testing and training regression for the used learning algorithms.

	Levenberg-Marquardt	Bayesian Regularization
Training fit	0.99992	0.99995
Testing fit	0.99991	0.99995
Validation fit	0.99992	-

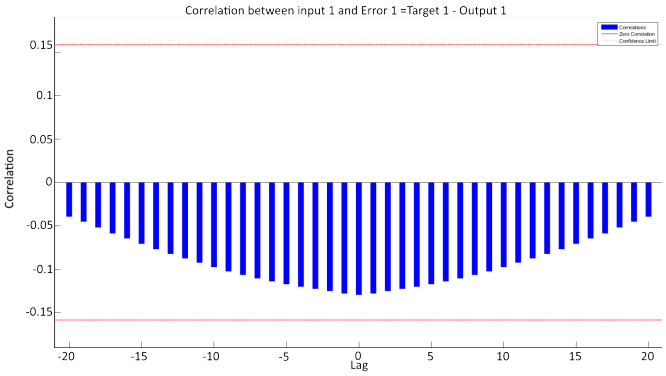
Table 3.5: Training Regression average values for both used learning algorithms

The represented regression values show that both testing and training data present good fitting to the target data for both learning algorithms. The Levenberg-Marquardt algorithm also uses a

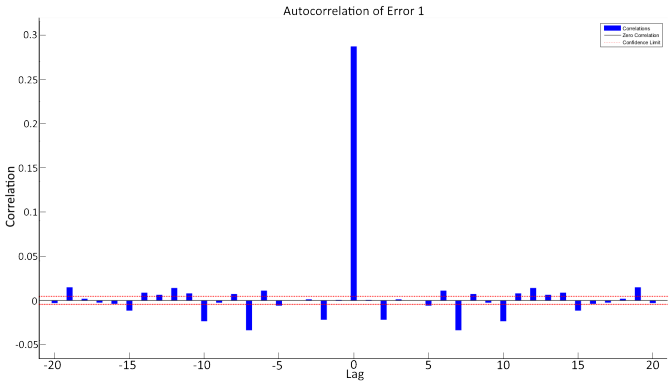
validation data set, hence its regression is also taken into consideration. The overall good fitting of different sets of data with targets are an indicator that overfitting did not occur [24].



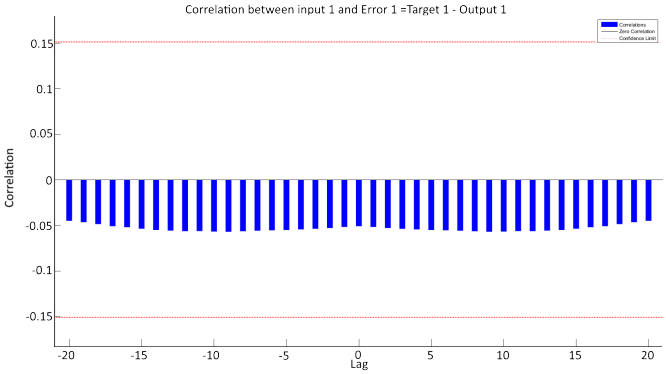
(a) Levenberg-Marquardt training error Autocorrelation



(b) Levenberg-Marquardt training error Correlation



(c) Bayesian Regularization training error Autocorrelation



(d) Bayesian Regularization training error Correlation

Figure 3.30: Network training Error Correlation and Autocorrelation

Error autocorrelation is a powerful performance indicator for the network, especially for prediction problems. For the prediction error to be uncorrelated, this function should be an impulse at zero. However, the algorithm only computes an estimate of the true autocorrelation function, therefore values for Lag≠0 will never be equal to zero. For that reason, confidence intervals are introduced to the plot, on which the correlation values must fall. In figures 3.30a and 3.30c, only a few number of Lag points fall off the confidence values which indicates that the error correlation is low for both training algorithms, thus validating the tapped delay lines imposed on the NARX model [24].

Error cross-correlation follows an analogous logic to Autocorrelation in terms of its analysis. In figures 3.30b and 3.30d, we can see that the cross-correlation values fall inside the confidence lines for all the Lag band, which indicates that no problematic is associated with this particular parameter.

The referred performance qualifiers tend to validate the training process. The parameter choice seems adequate for the approached problem, therefore all steps have been taken in order to acquire a well accomplished model for our generator behavior under working circumstances.

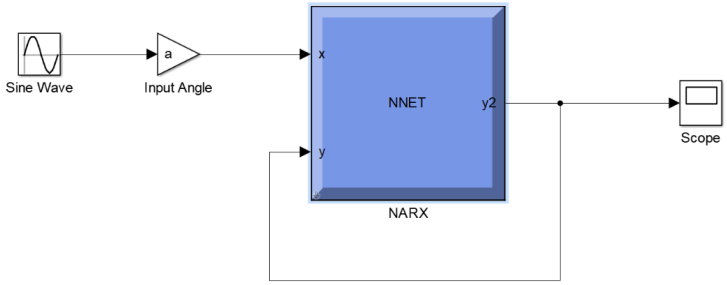
Since the simulation process of a network consists in a multiple step ahead prediction, the simulation must be performed in closed loop. This means that no real target data is provided to the network during simulation, therefore, it must perform multiple step ahead calculations based on previously calculated values. According to the developers of the used toolbox, close loop simulations often get better results when a posterior closed loop train is performed on the already trained

open loop network [35]. Therefore, this procedure was implemented in the designed structures in order to compare its results with the networks that have only been trained in open loop.

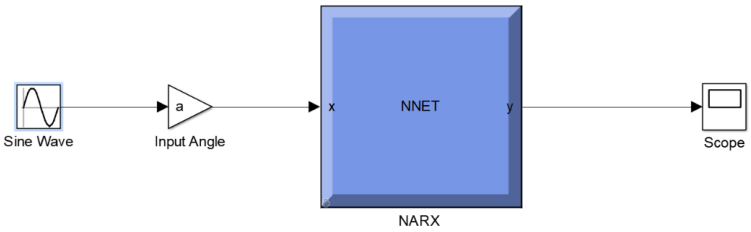
3.3.3 Network Simulation

After proper parametrization and training validation of the network, the developed model was simulated. This step is important for the experimental validation of the implemented network, which corresponds to the verification of its performance emulating the motion response by the magnet for various external excitation inputs. The network validation implies the replication of the studied system response to inputs that fall off the training data band. This means that a properly constructed and trained network structure should be able to approximate, with relative confidence, the dynamics of the generator for stimulations which features are not contained in the training set.

The trained network was simulated recurring to the Software Simulink®. The net structure, resultant from the script used to implement and train it, was converted into a block that contains all the features of the created network. This step was accomplished using the gensim(net,ts) function. The argument 'net' corresponds to the created network structure and 'ts' is the time step used by the network for each predicted value. From the generated NARX block, the simulation diagram is build. Figures 3.31a and 3.31b show two implementations of network blocks into diagrams, for simulation purposes.



(a) Network simulation for NARX block with intrinsic Open-loop features



(b) Network simulation for NARX block with intrinsic Closed-loop features

Figure 3.31: Possible training an simulation NARX Forms [24]

Figure 3.31a shows the simulation diagram for a NARX with intrinsic open-loop features. Figure 3.31b, shows the diagram of a NARX that has been trained in closed-loop, hence, its internal

configuration already possesses the closed-loop features required for simulation.

In order to properly synchronize network prediction values with the real data, the time step must be as close as possible to the acquisition sample time. The Arduino® controller records and prints one angle and one position value for each loop, and considering the simulation time for the training data acquisition as 200 seconds, the average sampling rate sr can be calculated, $ts = sr = \frac{Simulation\ time}{Number\ of\ data\ points}$. Since the input data vector contains 17039 elements, the resulting time step is of approximately 0.0117s. It is important to underline that this value is not dependent of the used simulation time, it is dependent of internal and external factors that affect the hardware, as well as the implemented code complexity. This means that for different simulations, regardless of its duration, the sampling rate should always be approximate.

The simulations outputs as well as their posterior analysis and experimental validation can be analyzed in chapter 4.

Chapter 4

Results

Regarding the results obtained in this work, two main subjects where taken into consideration.

The Networks testing results are presented and analyzed in order to determine an optimal configuration for final implementation.

The created generator power outputs are also matter of analysis in this chapter. Its study is important not only for applicability purposes but also to provide validation data regarding the functioning of the developed prototype.

4.1 Generator Performance

The estimation of the generator power outputs demanded the measurement of voltage drop between the terminals of a load resistance. From the Maximum Power Transfer theorem, it is known that a load with impedance equal to the source enables the maximization of power outputs [36]. Taking such considerations into view, the generator coils circuits were closed with a resistive load equivalent to each coil impedance, approximately 785Ω, performing a total resistance R_t of 1570Ω (considering a total resistive impedance for the source).

Inducing the generator with a specific stimulation during a defined amount of time provided the necessary data for the computing of Power outputs at specific conditions.

The generator was subjected to two distinct work condition waves, and the voltage values of all 4 coils where collected during T=10 seconds. The data was collected with a sampling rate of 0.05 using a dSpace DS1102. Figures 4.1 to 4.10 present plots of the generated Outputs values.

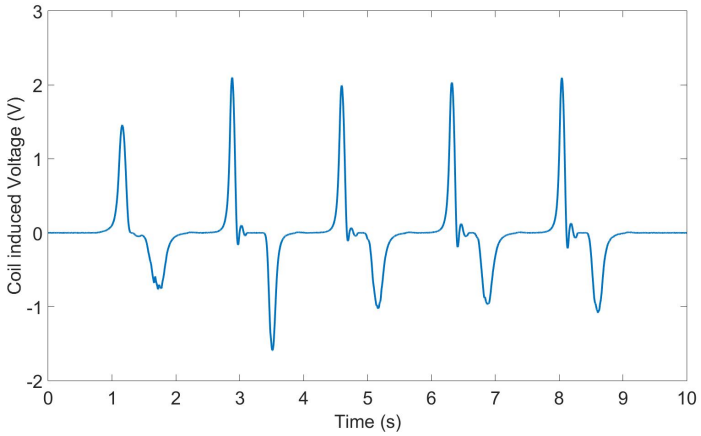


Figure 4.1: Coil 1 load resistance potential for a 20 degrees 0.5 Hz stimulation

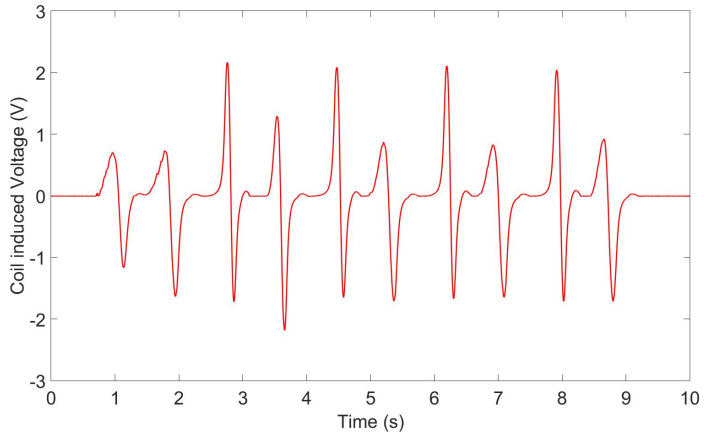


Figure 4.2: Coil 2 load resistance potential for a 20 degrees 0.5 Hz stimulation

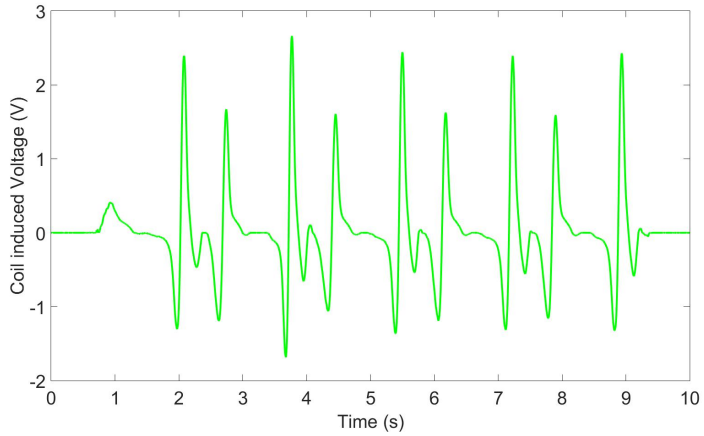


Figure 4.3: Coil 3 load resistance potential for a 20 degrees 0.5 Hz stimulation

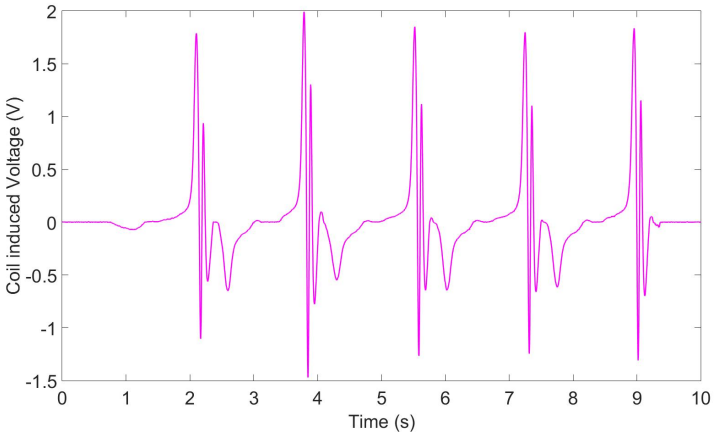


Figure 4.4: Coil 4 load resistance potential for a 20 degrees 0.5 Hz stimulation

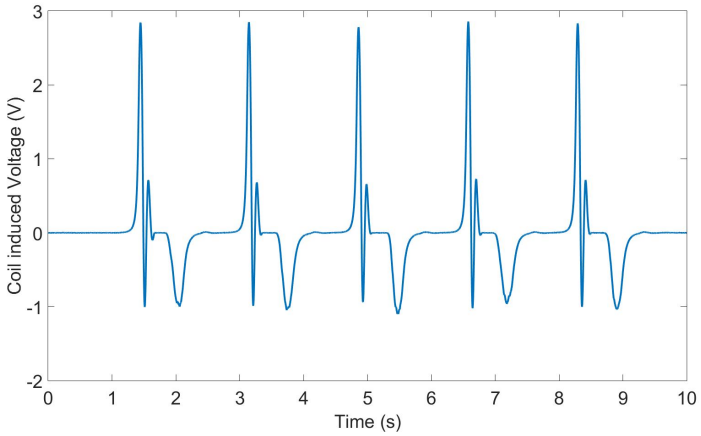


Figure 4.5: Coil 1 load resistance potential for a 25 degrees 0.5 Hz stimulation

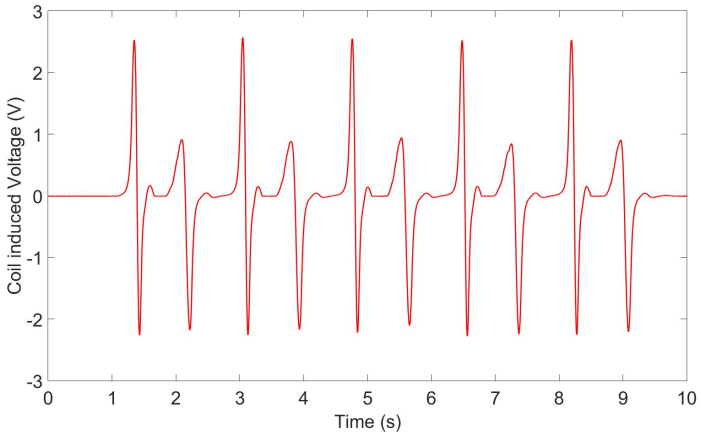


Figure 4.6: Coil 2 load resistance potential for a 25 degrees 0.5 Hz stimulation

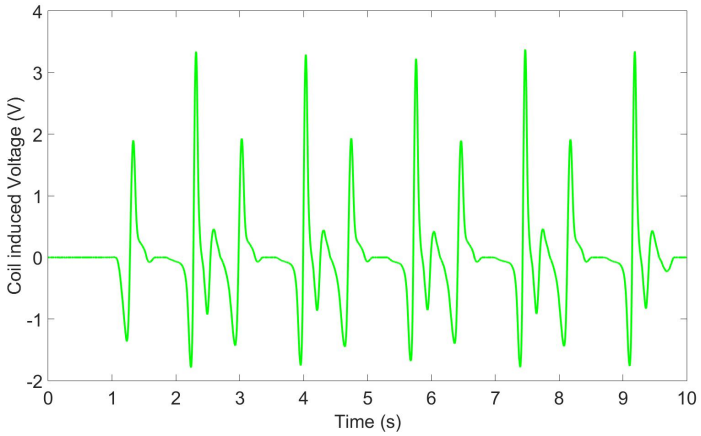


Figure 4.7: Coil 3 load resistance potential for a 25 degrees 0.5 Hz stimulation

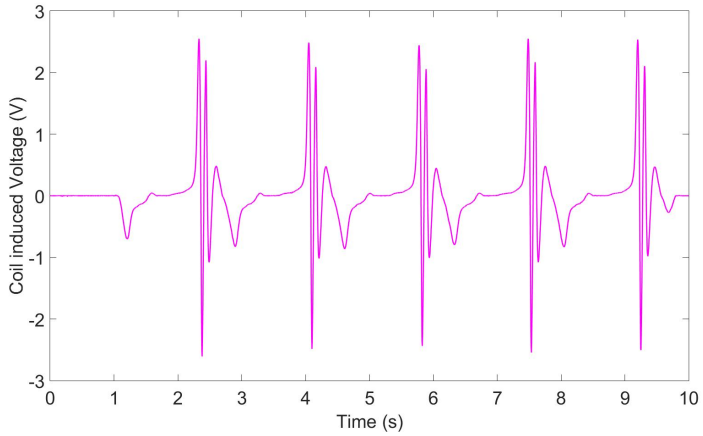


Figure 4.8: Coil 4 load resistance potential for a 25 degrees 0.5 Hz stimulation

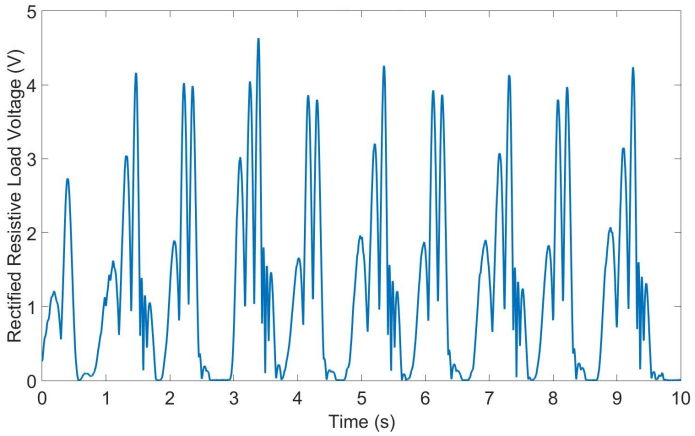


Figure 4.9: Rectified generated voltage from all coils at 20 degrees and 0.5 Hz stimulation

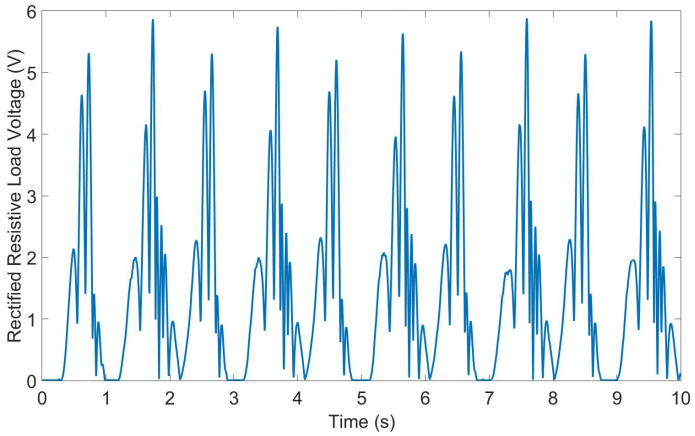


Figure 4.10: Rectified generated voltage from all coils at 25 degrees and 0.5 Hz stimulation

The respective data for each coil, for the two separate waves, was then used to compute the following output features:

- Load Current $i(t) = \frac{v(t)}{R_t}$
- Average Power $P_{avg} = \frac{1}{T}\int_0^T v(t).i(t).dt$
- Maximum instantaneous Power Output P_{max} =max$(v(t).i(t))$
- Energy generated during stimulation period $E = \int_0^T v(t).i(t).dt$

The computed values are presented in tables 4.1. These were calculated accounting for a rectified coil signal (module), also, the summation of different coil potential was carried out. This step was done to estimate the total output potential of the generator.

	P_{avg}(mW)	P_{max}(mW)	E(mJ)
Coil 1	0.438	12.4	4.4
Coil 2	0.73	12.3	7.3
Coil 3	0.66	11.6	6.6
Coil 4	0.237	5.7	2.4
All Coils	1.8	13.6	18.2

Table 4.1: Output values for stimulation at 20 degrees amplitude and 2 second period

	P_{avg}(mW)	P_{max}(mW)	E(mJ)
Coil 1	0.56	20.7	5.6
Coil 2	0.925	14.6	9.3
Coil 3	1.19	25.1	11.9
Coil 4	0.59	11.8	5.9
All Coils	2.6	22	26

Table 4.2: Output values for stimulation at 25 degrees amplitude and 2 second period

4.2 Network Testing

This section presents the results of the developed Networks for a wide variety of input stimulations. The Networks simulation for model validation and enhancement was performed according to three main topics:

- Comparison of outputs from two distinct learning algorithm based NARX
- Comparison between open-loop training results and open+closed-loop training results
- Comparison of the different models results for external excitations outside the training range

The NARX simulation consists in a determined number of consecutive step predictions. Since there are no target inputs during the intended simulation, this process must be done in a closed-loop configuration. As previously referred in chapter 3, for prediction optimization, it is often advised to perform a close-loop train to the already trained open-loop network. Considering two appropriate learning algorithms for the presented problem, it becomes relevant to correlate the effect of the loop configuration chosen for training with the two distinct learning algorithms. In

this section the evaluation of different training configuration is used to validate the implemented models and determine a suitable NARX model for all applications of the presented problem.

In order to perform a more reliable analysis to the results of each method, three networks where trained for each training configuration. Table 4.3 presents the implemented configurations as well as its nomenclature in posterior analysis.

	Training Architecture	
	Open Loop	Open and Closed Loop
Levenberg-Marquardt	$NARXLM_O$	$NARXLM_{OC}$
Bayesian Regularization	$NARXBR_O$	$NARXBR_{OC}$

Table 4.3: NARX training configurations

For this analysis, four specific excitation features where selected. One that is included in the training sample, and three that are not. The selected waves had the following characteristics:

	Amplitude (degrees)	Frequency (Hz)
Wave 1	35	0.8
Wave 2	22	1
Wave 3	20	0.67
Wave 4	28	0.67

Table 4.4: Network testing used wave configurations

The wave selection was made with view to cover a strategic set of possibilities. Wave 1 properties are contained in the training sample, Wave 2 frequency is also used in the training set but its amplitude is not, Wave 3 amplitude is covered in the training sample but not for that specific frequency and for Wave 4 neither of its parameters are included inside the training data. This has the purpose of determining if any specific configuration is better at generalizing for different inputs.

Figures 4.11 to 4.26 show the simulation results for the different proposed NARX configurations.

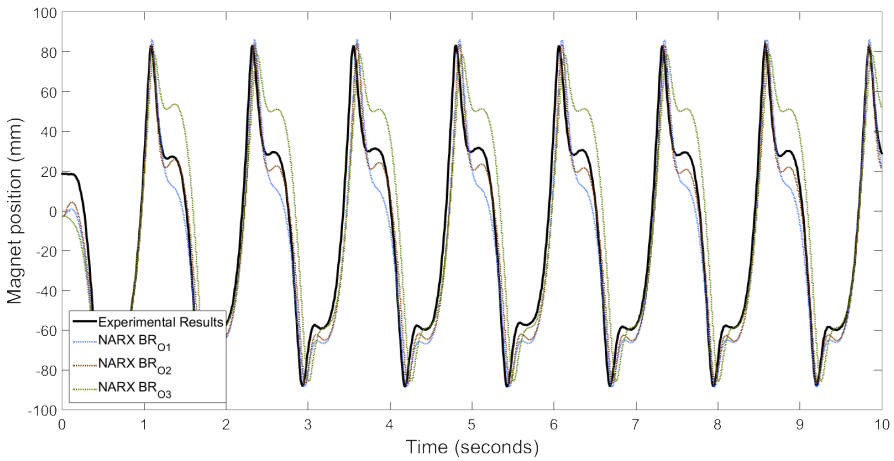


Figure 4.11: BR_O NARX simulation results for Wave 1

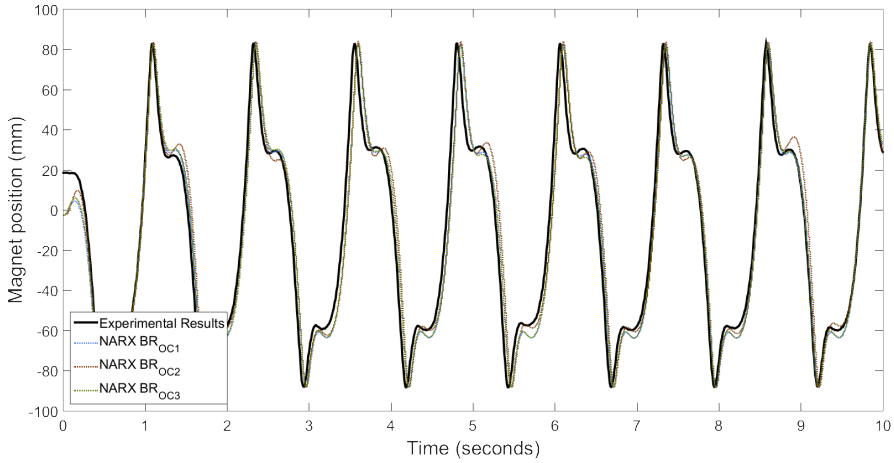


Figure 4.12: BR_{OC} NARX simulation results for Wave 1

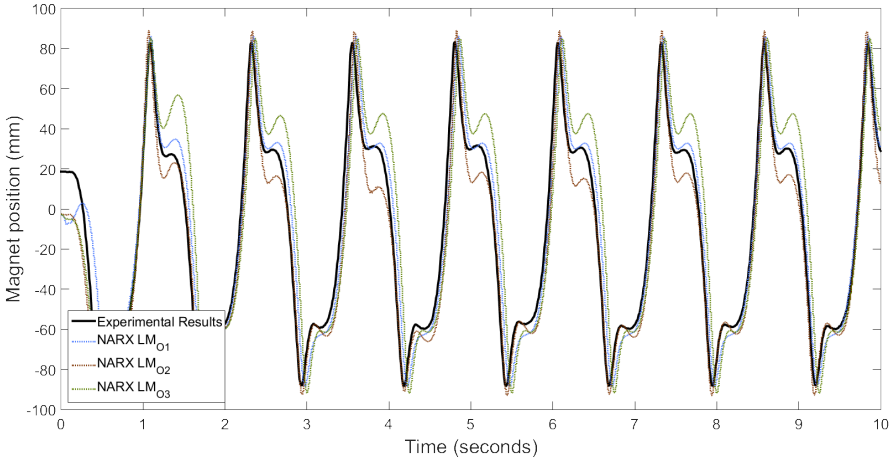


Figure 4.13: LM_O NARX simulation results for Wave 1

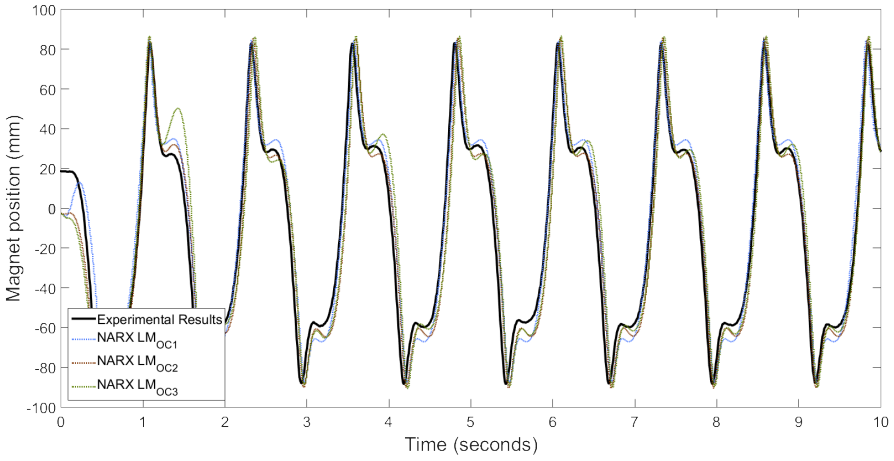


Figure 4.14: LM_{OC} NARX simulation results for Wave 1

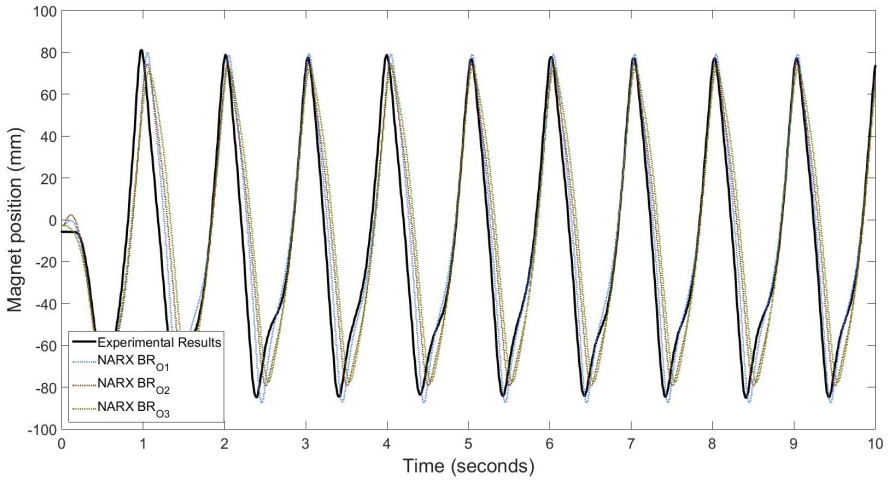


Figure 4.15: BR_O NARX simulation results for Wave 2

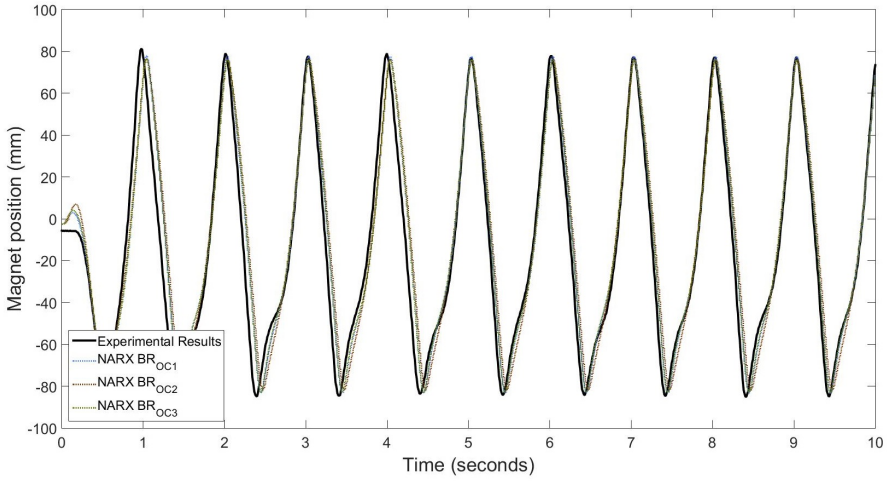


Figure 4.16: BR_{OC} NARX simulation results for Wave 2

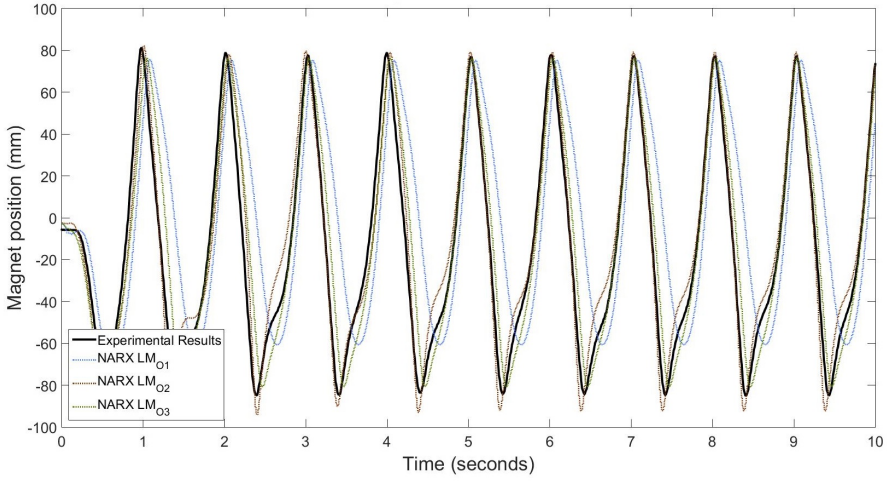


Figure 4.17: LM_O NARX simulation results for Wave 2

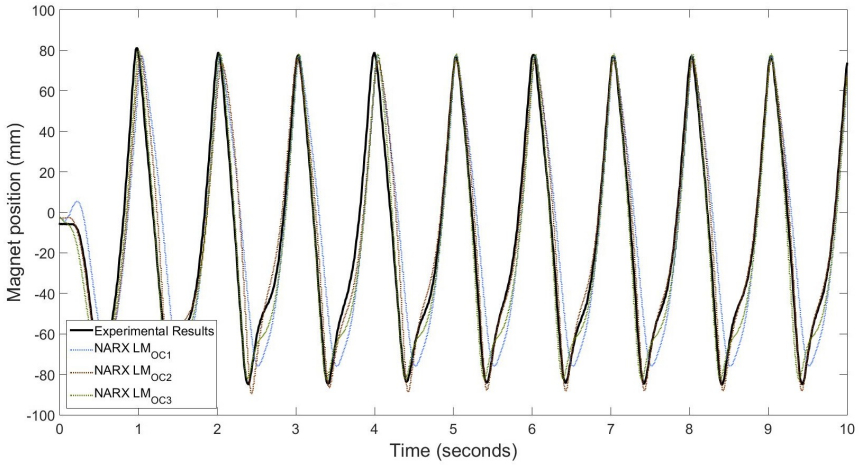


Figure 4.18: LM_{OC} NARX simulation results for Wave 2

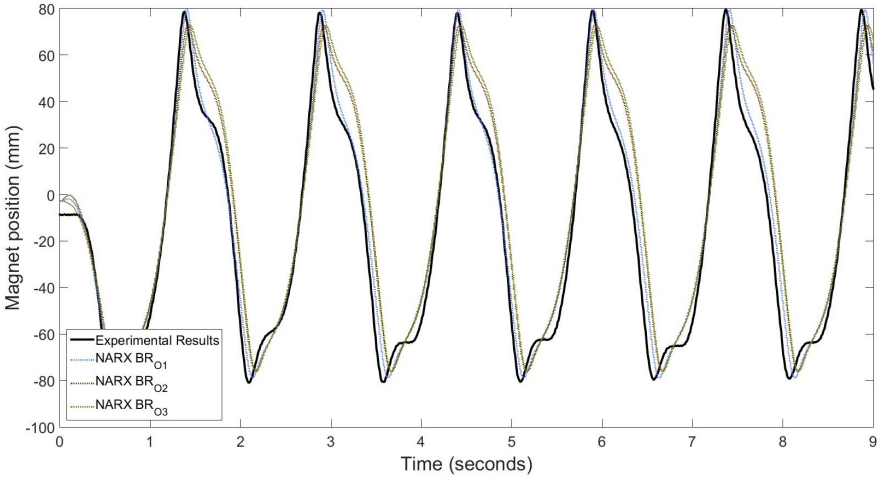


Figure 4.19: BR_O NARX simulation results for Wave 3

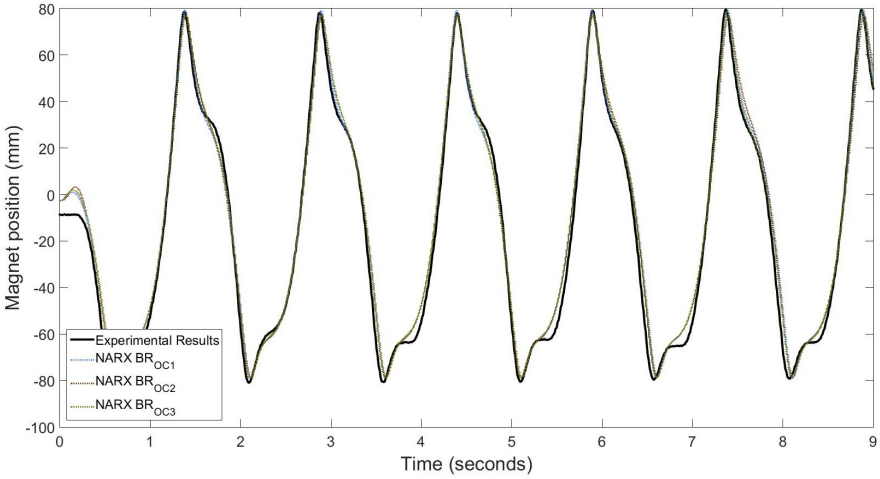


Figure 4.20: BR_{OC} NARX simulation results for Wave 3

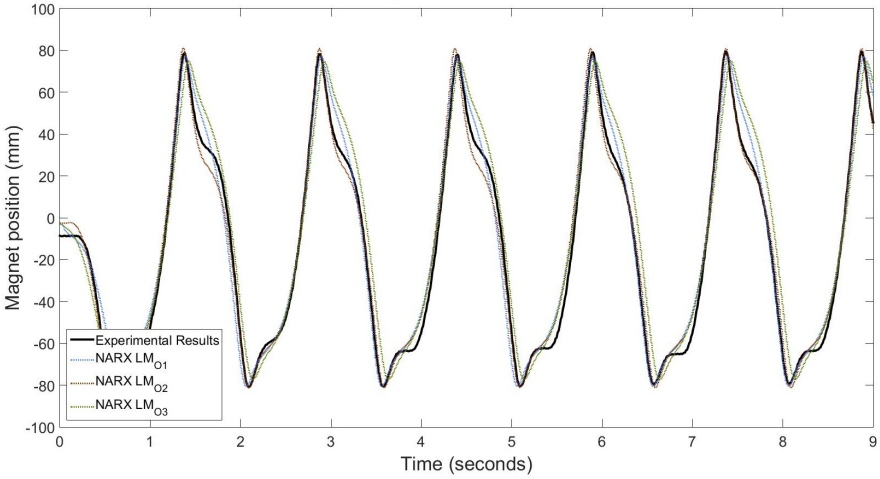


Figure 4.21: LM_O NARX simulation results for Wave 3

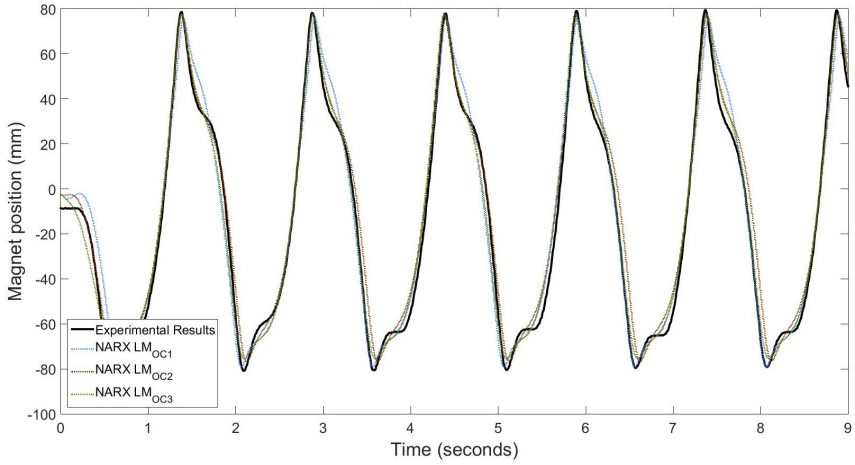


Figure 4.22: LM_{OC} NARX simulation results for Wave 3

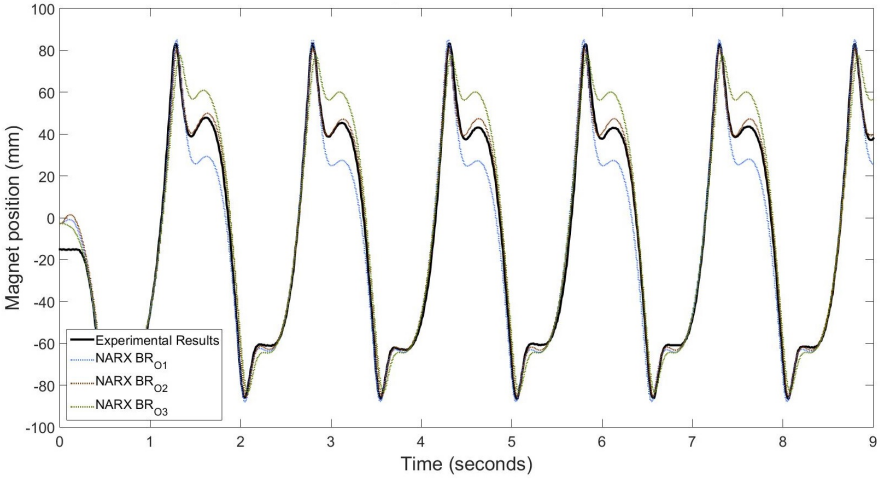


Figure 4.23: BR_O NARX simulation results for Wave 4

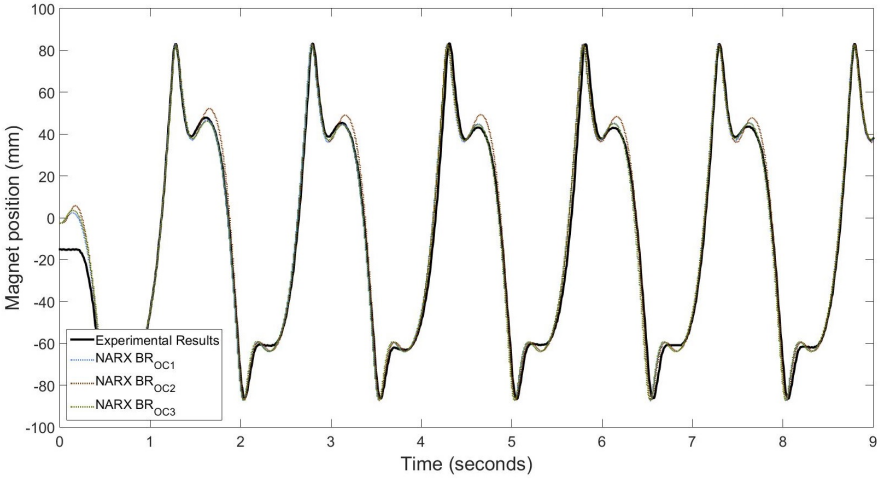


Figure 4.24: BR_{OC} NARX simulation results for Wave 4

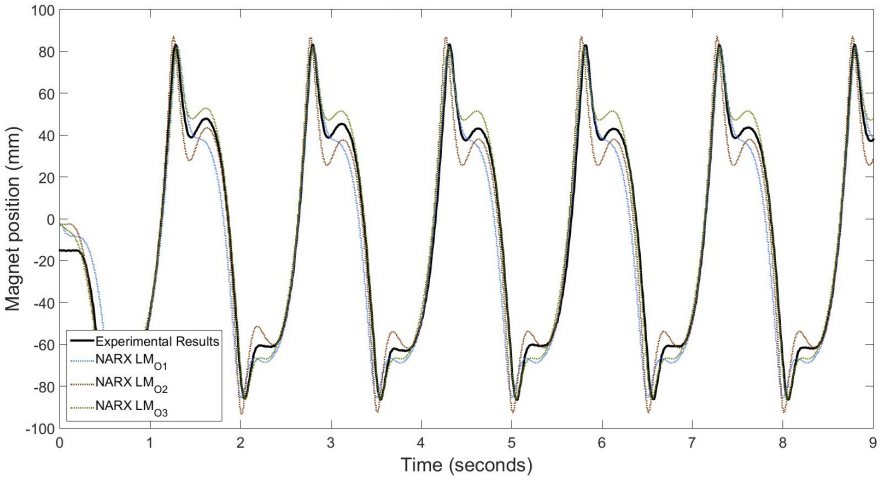


Figure 4.25: LM_O NARX simulation results for Wave 4

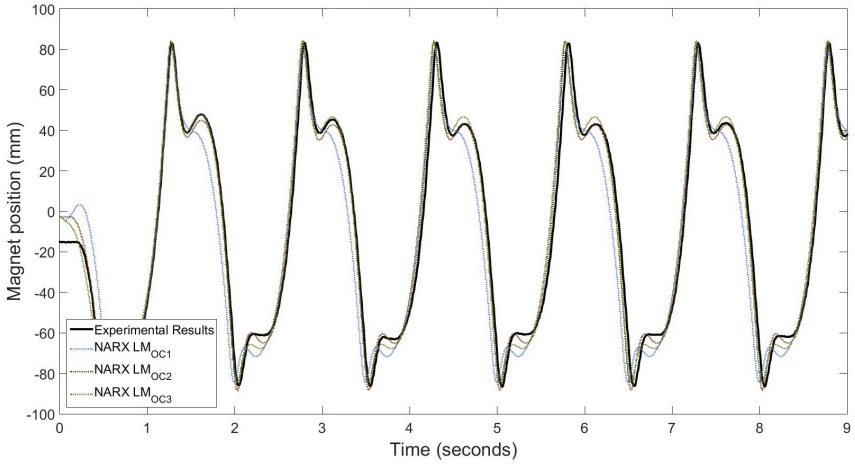


Figure 4.26: LM_{OC} NARX simulation results for Wave 4

For a proper validation of the created NARX structures and a more reliable choice of optimal configuration, the simulation results where submitted to a numerical evaluations. All the different network configurations where tested and compared to real experimental values in order to determine their generalization performance. For that matter three similarity indicators, mean squared error, correlation coefficient[5], and mean percentage error where calculated using immse() and xcorr() MATLAB® functions. The xcorr() function was used to obtain the correlation coefficient at lag 0, which is presented in percentage for a more intuitive analysis. As for mean percentage error, it was computed using equation 4.1,

$$MPE = \frac{100}{n} \sum_{t=1}^{n} \frac{T_n - P_n}{T_n} \tag{4.1}$$

where T_n and P_n correspond to target and predicted series respectively. The accuracy descriptors computing script can be seen in figures B.6 and B.7, in annexed documents.

	MSE		CC		MPE	
	Average	Minimum	Average	Maximum	Average	Minimum
$NARXBR_O$	196.65	149.19	94.46%	97.04%	31.27%	11.17%
$NARXBR_{OC}$	145.32	127.66	97.19%	97.55%	24.12%	14.18%
$NARXLM_O$	302.85	132.99	94.38%	97.43%	38.59%	2.91%
$NARXLM_{OC}$	192.60	117.19	96.34%	97.74%	31.51%	22.13%

Table 4.5: Simulation accuracy descriptors for NARX with different training configurations (Wave 1)

	MSE		CC		MPE	
	Average	Minimum	Average	Maximum	Average	Minimum
$NARXBR_O$	412.85	167.81	91.87%	96.75%	25.60%	20.15%
$NARXBR_{OC}$	142.41	113.02	97.24%	97.82%	18.75%	16.87%
$NARXLM_O$	648.75	111.59	86.60%	97.85%	24.42%	14.48%
$NARXLM_{OC}$	214.13	78.04	95.87%	98.60%	18.19%	13.43%

Table 4.6: Simulation accuracy descriptors for NARX with different training configurations (Wave 2)

	MSE		CC		MPE	
	Average	Minimum	Average	Maximum	Average	Minimum
$NARXBR_O$	164.49	47.76	96.80%	99.10%	27.82%	23.81%
$NARXBR_{OC}$	27.63	24.60	99.52%	99.58%	26.89%	21.19%
$NARXLM_O$	69.28	35.49	98.68%	99.33%	30.85%	5.99%
$NARXLM_{OC}$	45.37	30.79	99.15%	99.41%	36.79%	12.01%

Table 4.7: Simulation accuracy descriptors for NARX with different training configurations (Wave 3)

	MSE		CC		MPE	
	Average	Minimum	Average	Maximum	Average	Minimum
$NARXBR_O$	65.63	18.87	98.86%	99.60%	9.62%	1.56%
$NARXBR_{OC}$	42.64	33.86	99.22%	99.39%	6.10%	0.76%
$NARXLM_O$	102.63	33.43	98.17%	99.48%	8.25%	1.9%
$NARXLM_{OC}$	121.84	48.65	97.80%	99.11%	15.66%	5.32%

Table 4.8: Simulation accuracy descriptors for NARX with different training configurations (Wave 4)

The light green cells indicate the best NARX results for that specific indicator. From an overall analysis of the table, the verification that the prediction made by the Network tends to be more accurate outside training range values is a good indicator for the NARX structure quality [24]. A comprehensive reading of the Cross-correlation coefficient also indicates that globally, all Networks present reasonably good accuracy relatively to real experimental values.

A deeper analysis to the simulation descriptors leads to the conclusion that the BR_{OC} structure is the more consistent training configuration regarding best performance attainment. Although other NARX configurations occasionally present best average or minimum/maximum values for a specific descriptor, the referred configuration presents better generalization features inside and outside training range.

Taking such conclusion into consideration, the remaining step consists in a deeper validation of this model through further simulation. The best performed Network among the trained $NARXBR_{OC}$ was the object for this validation.

The wave configurations used for final validation can be seen in the following table 4.9.

Frequency (Hz)	Amplitude 1 (degrees)	Amplitude 2 (degrees)	Amplitude 3 (degrees)	Amplitude 4 (degrees)	Amplitude 5 (degrees)
1	17	22	28	35	38
0.8					
0.67					
0.5					
0.4					

Table 4.9: Wave configurations for final testing

Adding to the constant sinusoidal waves, four wave configurations resultant from wave summation where also tested in the network. These where implemented in order to verify the response of the network for more complex inputs. The applied waves formulation can be seen in the following table 4.10. For nomenclature simplification, these waves where identified as complex waves (CWaves).

	Wave formulation
CWave1	$15\sin(\frac{2\pi}{3}t)+30\sin(\frac{2\pi}{1.5}t)$
CWave2	$15\sin(\frac{2\pi}{2.6}t)+30\sin(\frac{2\pi}{1.5}t)$
CWave3	$10\sin(\frac{2\pi}{3}t)+30\sin(2\pi t)$
CWave4	$30\sin(\frac{2\pi}{6}t)+30\sin(\frac{2\pi}{2.5}t)$

Table 4.10: Complex wave configurations for final testing

Figures 4.27 to 4.55 presents BR_{OC} NARX results for all simulation conditions.

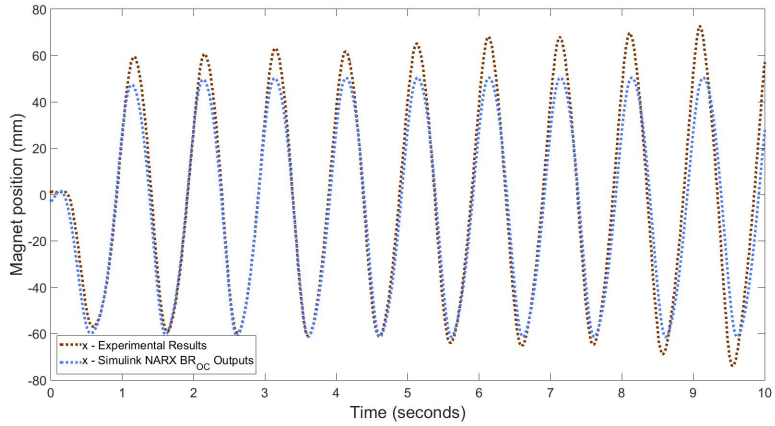


Figure 4.27: Network Outputs vs Experimental results for wave with 17 degrees amplitude and 1 Hz (CC=97.48%)

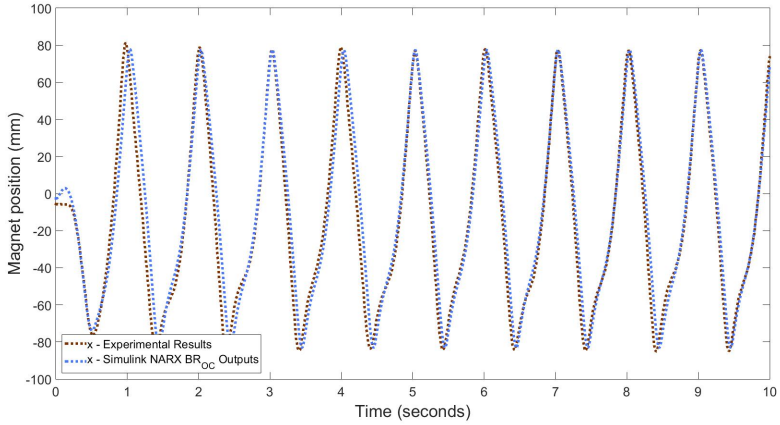


Figure 4.28: Network Outputs vs Experimental results for wave with 22 degrees amplitude and 1 Hz (CC=97.73%)

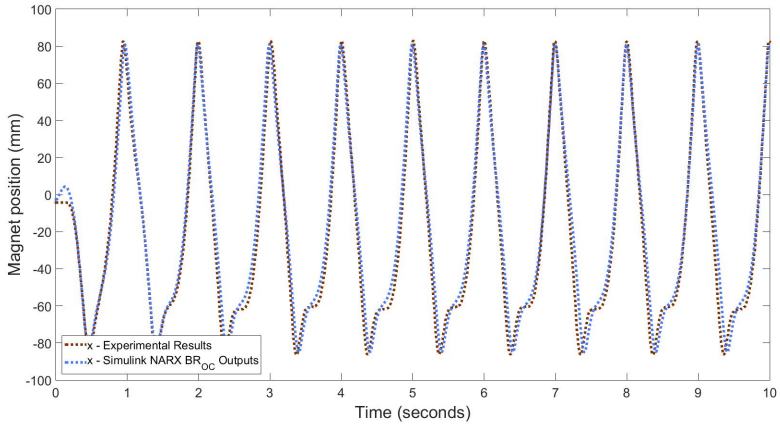


Figure 4.29: Network Outputs vs Experimental results for wave with 28 degrees amplitude and 1 Hz (CC=99.07%)

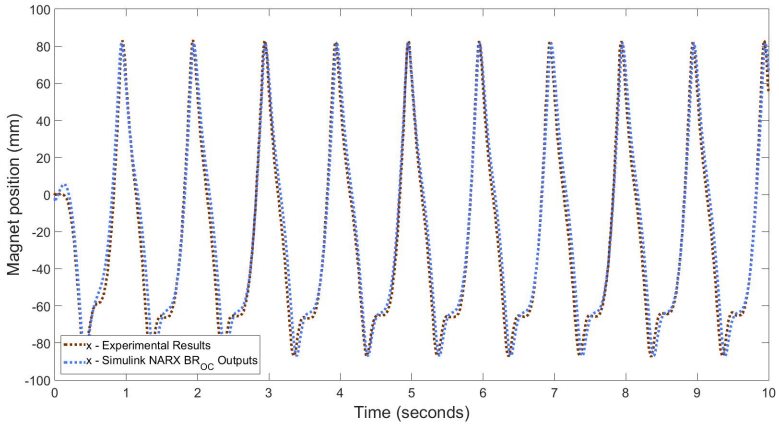


Figure 4.30: Network Outputs vs Experimental results for wave with 35 degrees amplitude and 1 Hz (CC=98.82%)

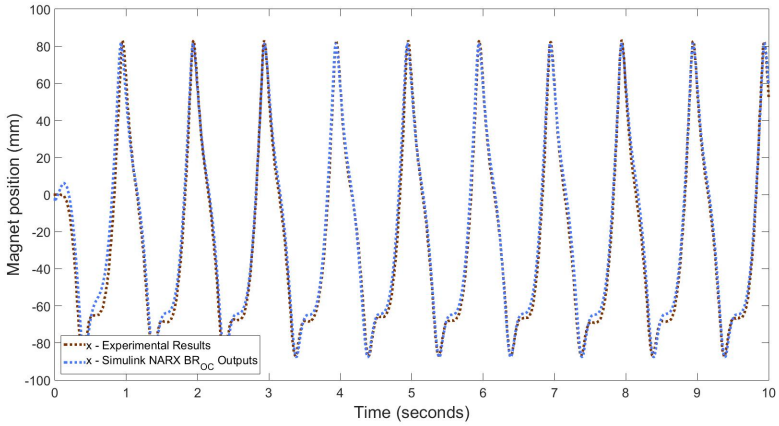


Figure 4.31: Network Outputs vs Experimental results for wave with 38 degrees amplitude and 1 Hz (CC=99.52%)

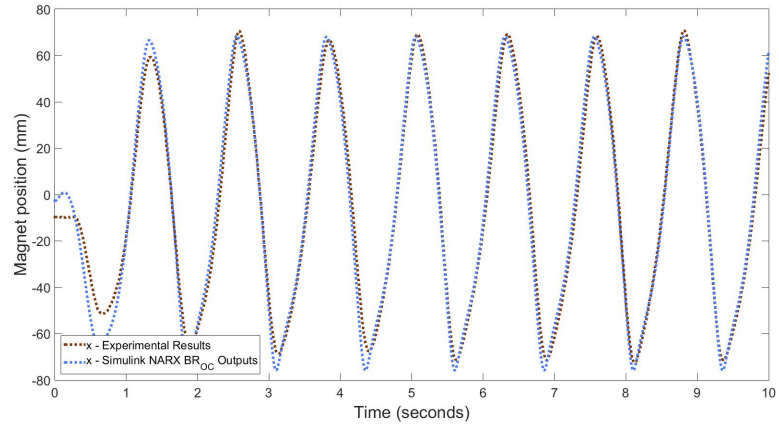


Figure 4.32: Network Outputs vs Experimental results for wave with 17 degrees amplitude and 0.8 Hz (CC=99.22%)

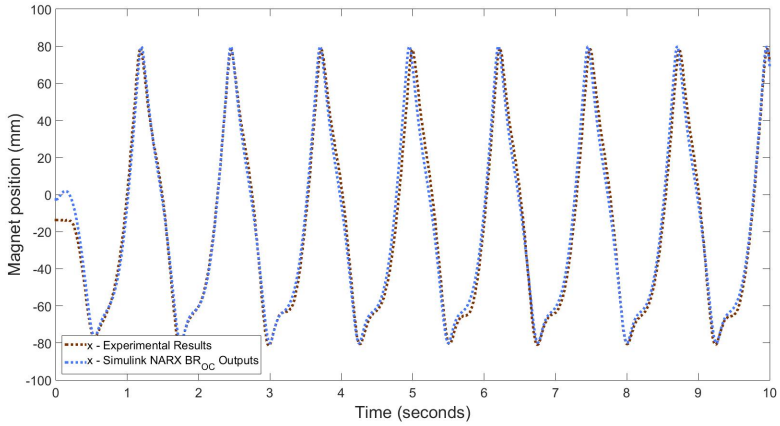


Figure 4.33: Network Outputs vs Experimental results for wave with 22 degrees amplitude and 0.8 Hz (CC=99.16%)

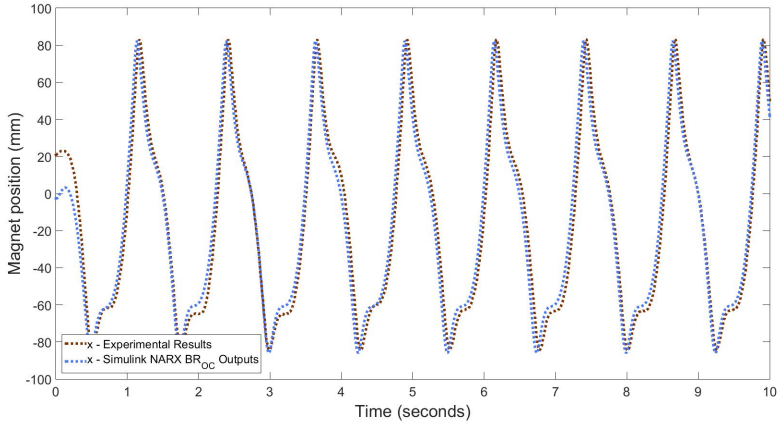


Figure 4.34: Network Outputs vs Experimental results for wave with 28 degrees amplitude and 0.8 Hz (CC=98.00%)

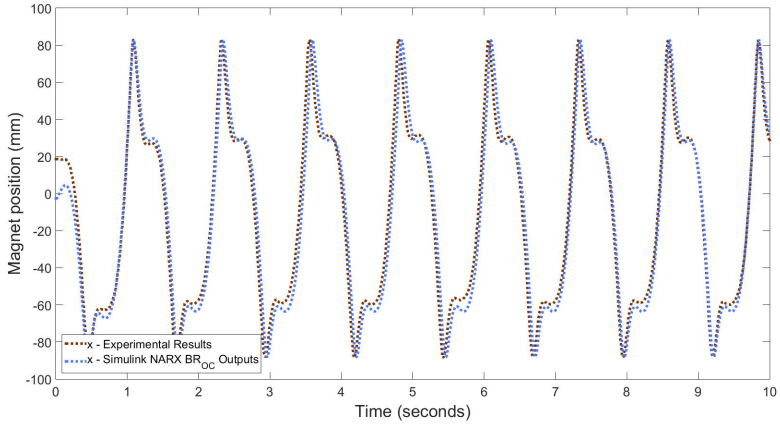


Figure 4.35: Network Outputs vs Experimental results for wave with 35 degrees amplitude and 0.8 Hz (CC=98.24%)

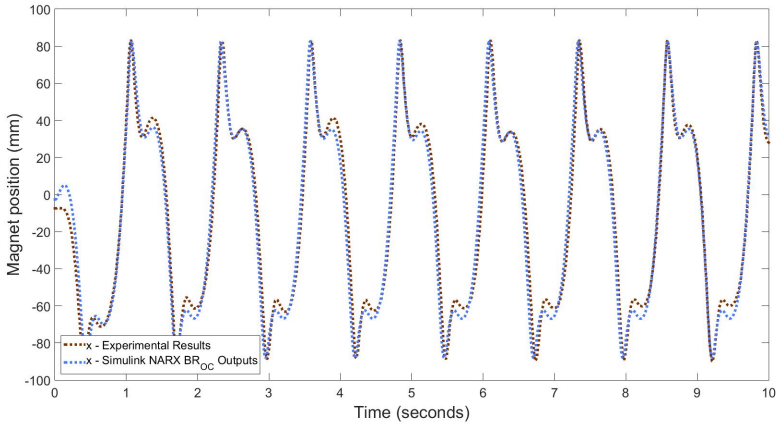


Figure 4.36: Network Outputs vs Experimental results for wave with 38 degrees amplitude and 0.8 Hz (CC=98.90%)

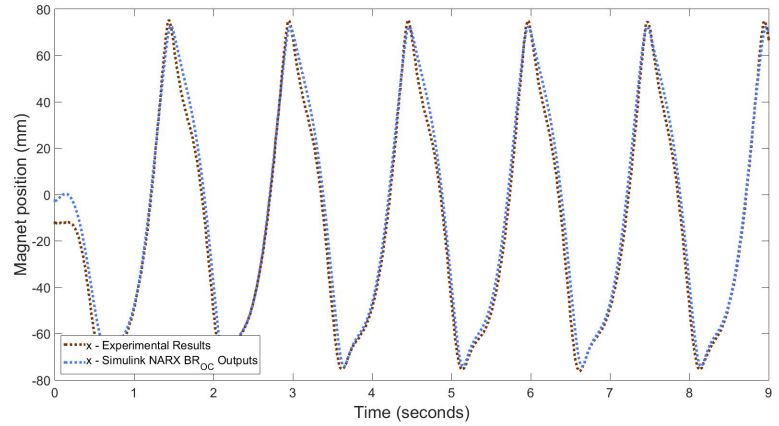


Figure 4.37: Network Outputs vs Experimental results for wave with 17 degrees amplitude and 0.67 Hz (CC=99.20%)

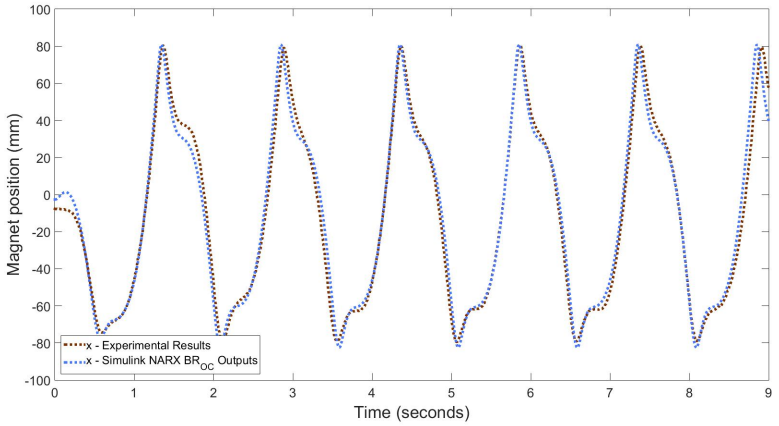


Figure 4.38: Network Outputs vs Experimental results for wave with 22 degrees amplitude and 0.67 Hz (CC=99.02%)

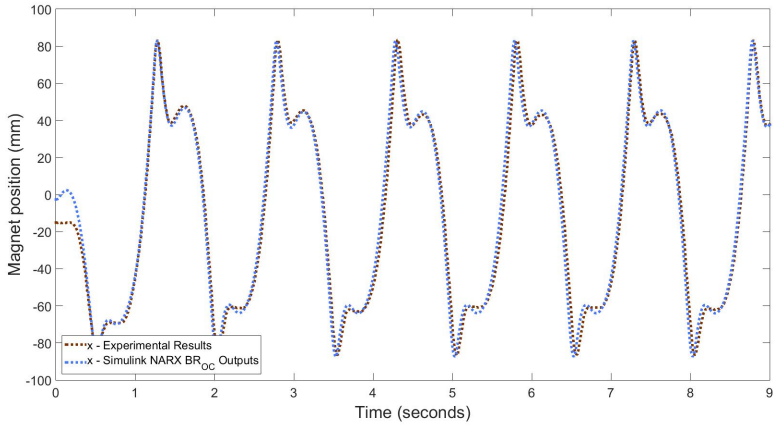


Figure 4.39: Network Outputs vs Experimental results for wave with 28 degrees amplitude and 0.67 Hz (CC=99.12%)

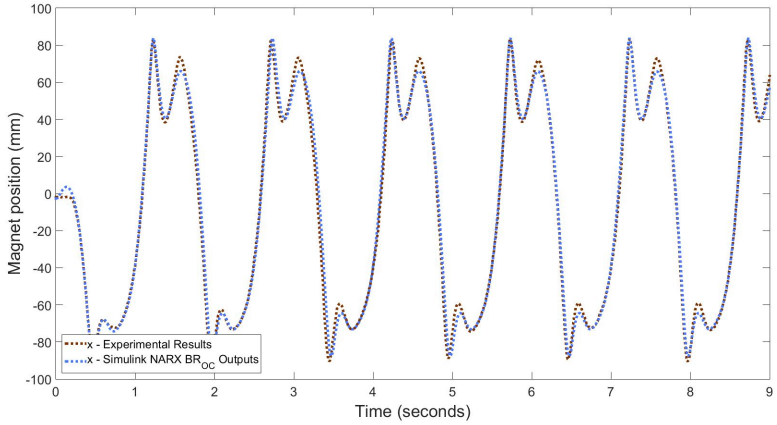


Figure 4.40: Network Outputs vs Experimental results for wave with 35 degrees amplitude and 0.67 Hz (CC=99.69%)

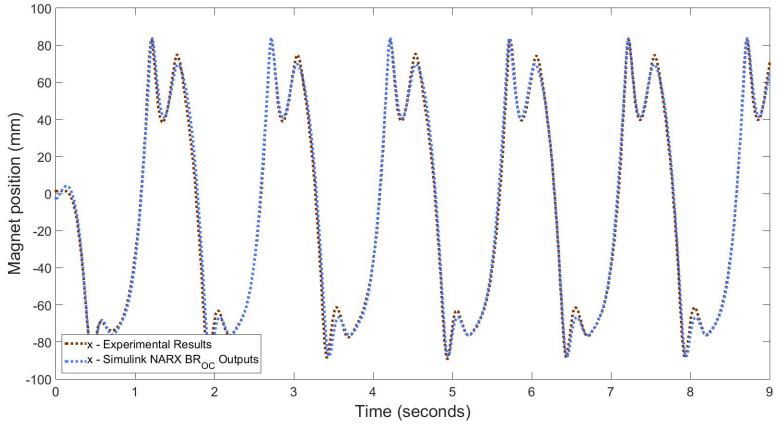


Figure 4.41: Network Outputs vs Experimental results for wave with 38 degrees amplitude and 0.67 Hz (CC=99.72%)

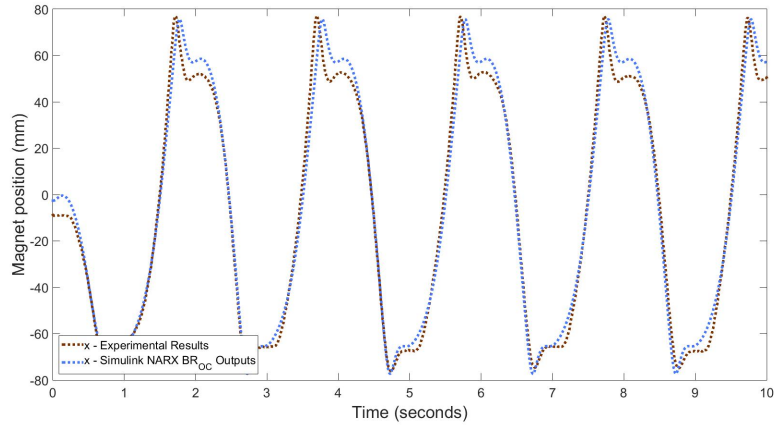


Figure 4.42: Network Outputs vs Experimental results for wave with 17 degrees amplitude and 0.5 Hz (CC=99.20%)

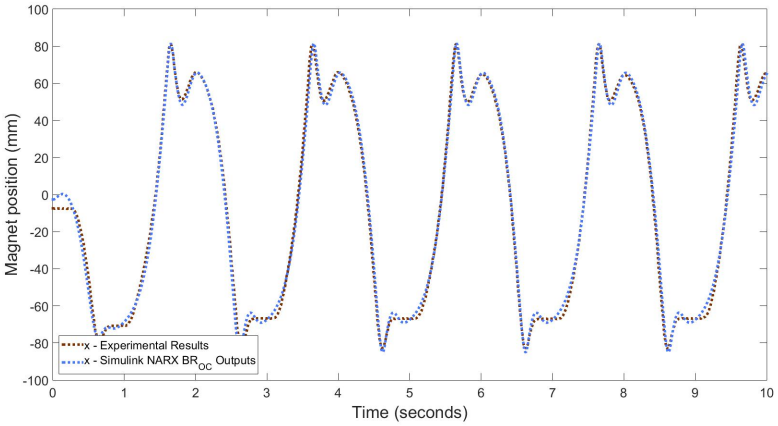


Figure 4.43: Network Outputs vs Experimental results for wave with 22 degrees amplitude and 0.5 Hz (CC=99.82%)

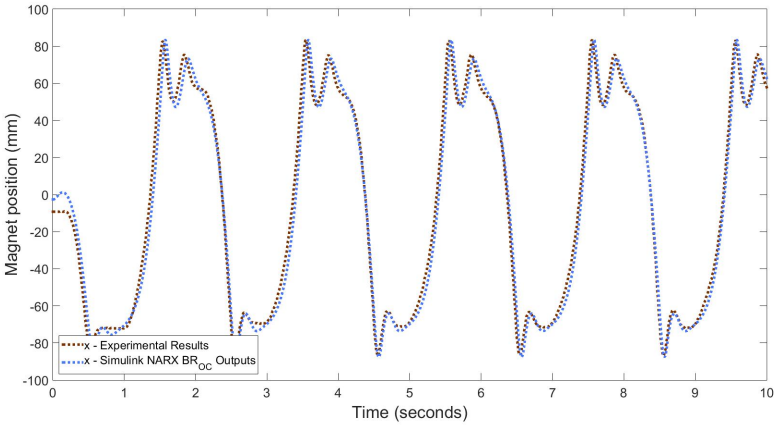


Figure 4.44: Network Outputs vs Experimental results for wave with 28 degrees amplitude and 0.5 Hz (CC=99.25%)

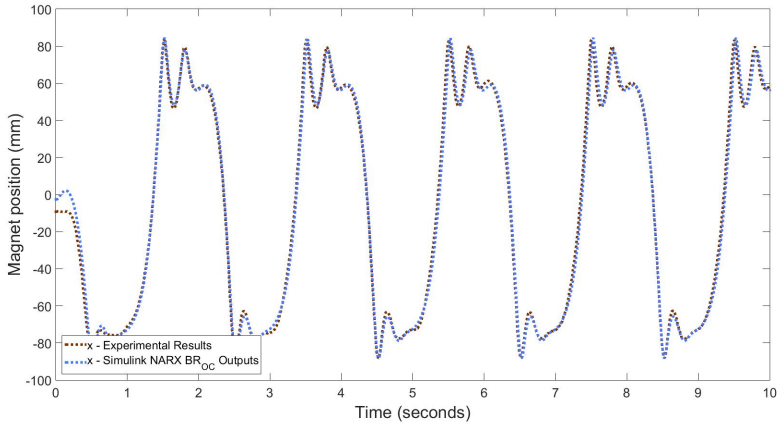


Figure 4.45: Network Outputs vs Experimental results for wave with 35 degrees amplitude and 0.5 Hz (CC=99.74%)

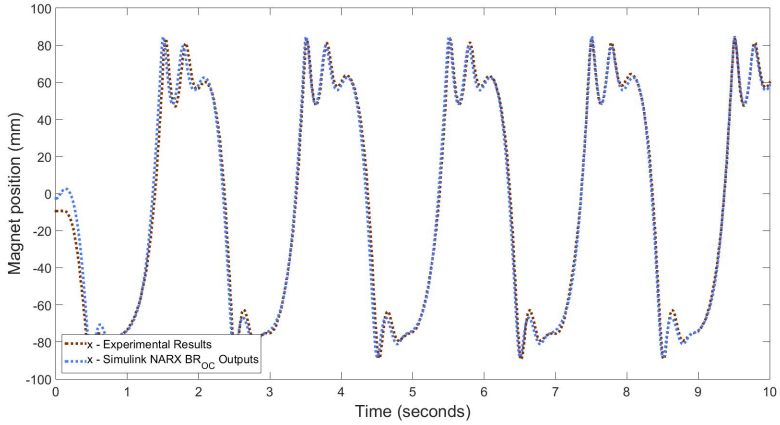


Figure 4.46: Network Outputs vs Experimental results for wave with 38 degrees amplitude and 0.5 Hz (CC=99.56%)

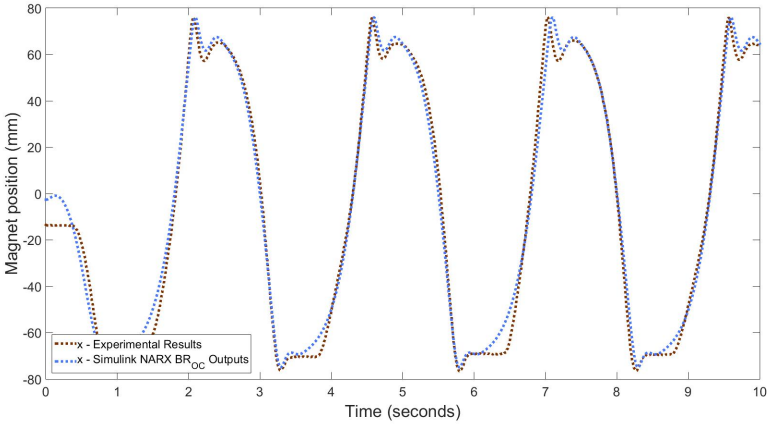


Figure 4.47: Network Outputs vs Experimental results for wave with 17 degrees amplitude and 0.4 Hz (CC=99.54%)

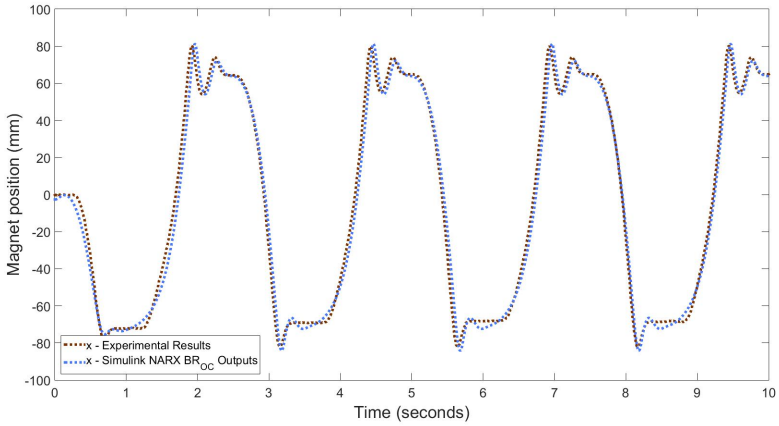


Figure 4.48: Network Outputs vs Experimental results for wave with 22 degrees amplitude and 0.4 Hz (CC=99.47%)

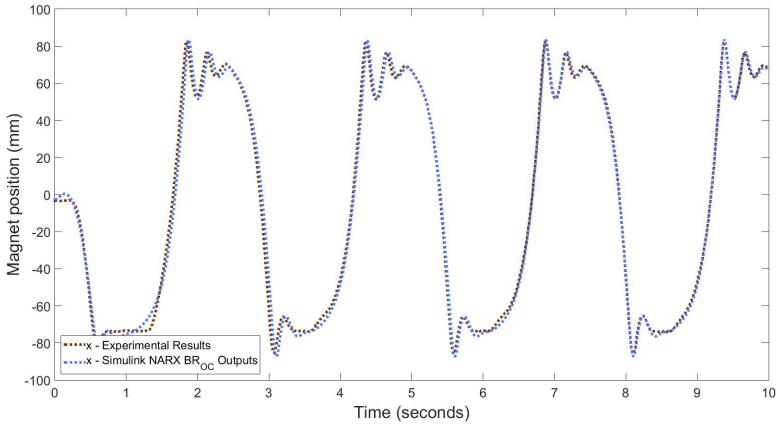


Figure 4.49: Network Outputs vs Experimental results for wave with 28 degrees amplitude and 0.4 Hz (CC=99.78%)

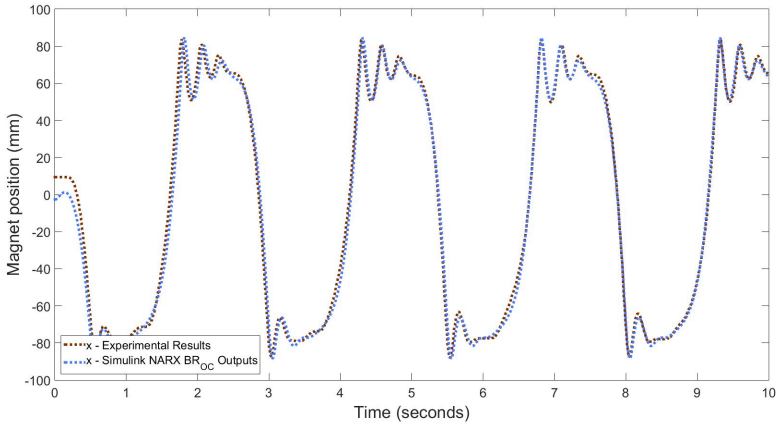


Figure 4.50: Network Outputs vs Experimental results for wave with 35 degrees amplitude and 0.4 Hz (CC=99.66%)

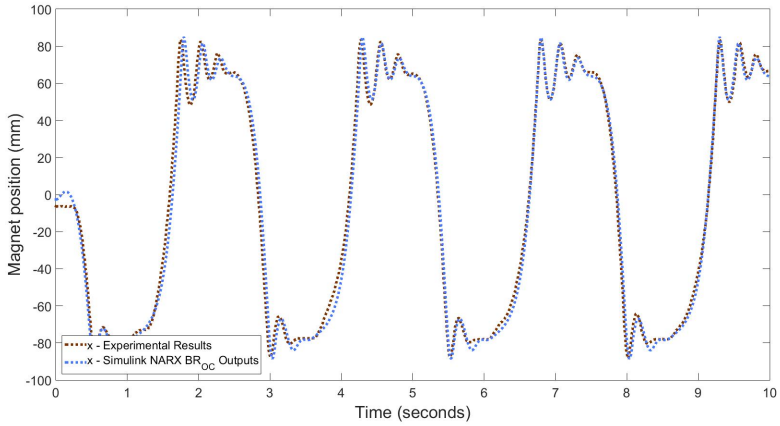


Figure 4.51: Network Outputs vs Experimental results for wave with 38 degrees amplitude and 0.4 Hz (CC=99.58%)

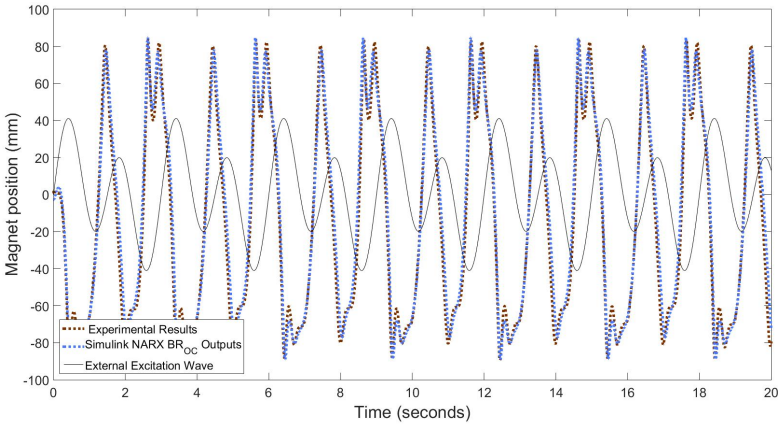


Figure 4.52: Network Outputs vs Experimental results for CWave 1 (CC=98.92%)

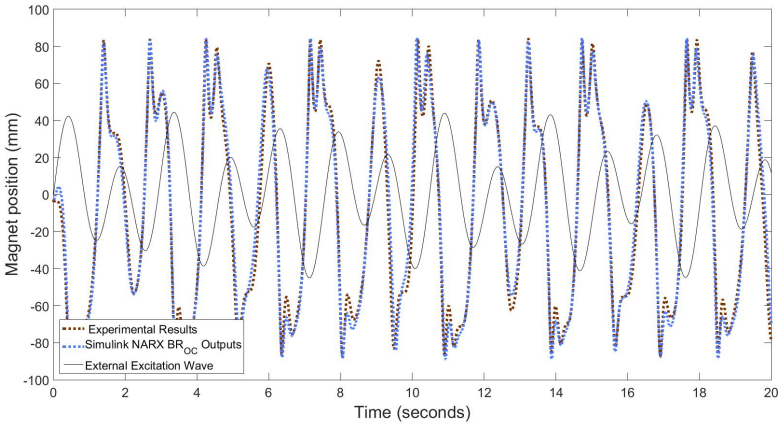


Figure 4.53: Network Outputs vs Experimental results for CWave 2 (CC=99.18%)

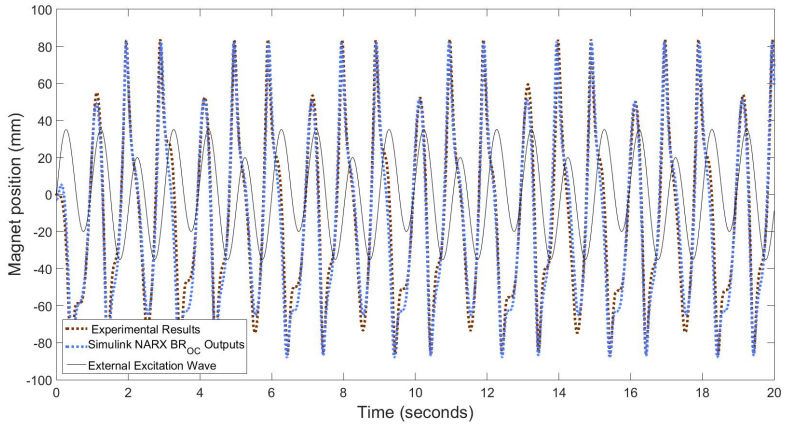


Figure 4.54: Network Outputs vs Experimental results for CWave 3 (CC=97.98%)

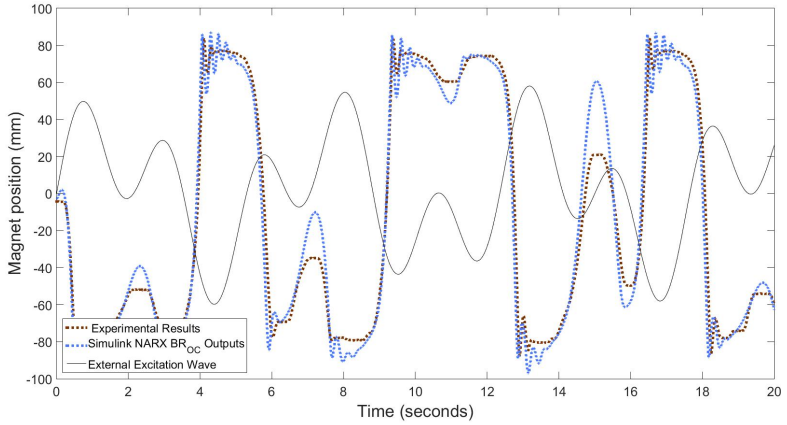


Figure 4.55: Network Outputs vs Experimental results for CWave 4 (CC=98.43%)

Chapter 5

Discussion and Conclusions

LBEH prototype

From the design of the prototype to its construction and testing, several aspects have been noticed and pointed for future enhancements to posterior models. In terms of overall performance it is safe to say that the developed device does fulfill most of the proposed desired characteristics. Hence, it seems correct to assume that this device represents a good approximation to a pretended applicable energy harvester. However, some aspects must be reviewed to prevent operation flaws, retard degradation and optimize outputs.

Regarding experimental behavior issues, the most noticeable flaw lies in the usage of a metallic linear bearing. The magnetic influence on the metallic balls, although weak and unnoticeable in the first tests, tends to hinder the correct functioning of the bearing. Also, from visualization of the experiments, it is concluded that the friction coefficient between the acrylic and the magnet is also an optimization issue to take into consideration. Another experimentally verified issue is the weak repulsion force exerted by the fixed magnets, which results in impact of the magnet array on the caps if the input angle amplitude is too large. This phenomenon also results in magnetic attraction when repulsing magnets make contact, locking the levitating magnet array in one of the extremities of the device when this occurs.

The overall configuration of the generator still cares for some optimization. However, from the outputs analysis and behavior, the developed model can be considered applicable for the intended purposes.

LBEH modeling using ANN

The modeling of an electromagnetic levitation generator provides key information for the optimization of the device and prediction of its energy scavenging potentialities for determined working environment features.

The problem approach using machine learning techniques has proven its potentialities not only from the results quality perspective but also from a practical point of view. Besides providing accurate prediction results for the model, using NARX allows for the modeling of a system that would reveal very complex and time consuming when solved using Analytical, Semi-Analytical or even other Numerical approaches. The method presented in this work, enables the obtainment of the motion response over time to a determined input stimulation using nothing but recorded data from the very same, while other methods require the formulation of all the complex subsystems that are presented in a non-linear dynamic system. The developed NARX model also allows for simulations with input outside the training data range.

Beside from this, behavior predictions using the approach taken in this work, are constrained to features intrinsic to the generator used for data collecting. This presents a limitation when

comparing to approaches taken in the literature, where it is possible to vary parameter that are directly related to the harvesting device, like damping or friction coefficients.

Regarding computation time, this technique excels. Since a properly trained network, which takes about 5 minutes to train when considering the best suiting configuration, can compute the behavior of the magnet for a specific external excitation in approximately 1 second.

5.1 Conclusions and future work

The results presented in this work justify the proposed objective achievement. The development of a LBEH was performed, and, according to output results and overall device performance evaluation, it seems fair to state its applicability. The testing station also complied the intended purpose, since it enabled the stimulation of the device and synchronous data acquisition.

A Neural Network was developed through the course of this work, the NARX structure did perform according to expectation and its applicability on non linear system prediction verified.

Regarding the NARX training for system modeling, the possibility to predict the modeled system behavior with relative precision for a desired stimulation, proves the potential of this methodology in applications such as the one in this work. According to simulation results, it seems reasonable to affirm that the modeling method used in this work is suitable for the application, thus permitting the acquisition of useful information for future performance optimization.

Naturally, as result of a general prototype analysis, the realization that the physical model still cares for a few enhancements becomes noticeable. Besides from the need of a working platform for intended applications, such as a buoy if applied for sea motion transduction, the prototype possesses a few topics that must be reviewed before development of applicable models. Depending on the specific application, the mobile components of the generator could still use some improvements. In order to improve the motion of the magnet inside the generator, more than one strategy can be used. The use of a second guiding shaft for the levitating magnet would significantly reduce friction between magnet and the inner walls. However this approach will confer a less compact architecture, as well as enhancing the chances of component failure and degradation. In case this methodology is applied, the appliance of linear bearings with polymer balls instead of metallic ones can prevent degradation due to oxidation, and avoid the interaction of the magnetic field in the bearing performance. In case the use of a guiding shaft is discarded, the use of an inner surface with less friction than acrylic is advised. The alteration of fixed magnets configuration is also required in future model development. More than one magnet for each cap seems necessary, since the repulsive force of a single magnet per cap appears to be insufficient to prevent impact in high amplitude stimulations.

The experimental data collected from the developed model, as well as the information provided by the ANN modeled system can be very useful for future performance optimization. Therefore, the development of a new prototype based on the presented model and enhanced according to the experimental and theoretical information provided by this work is suggested as future work.

Bibliography

[1] Raul Morais et al. “Double permanent magnet vibration power generator for smart hip prosthesis”. In: *Sensors and Actuators, A: Physical* 172.1 (2011), pp. 259–268. ISSN: 09244247. DOI: 10.1016/j.sna.2011.04.001. URL: http://dx.doi.org/10.1016/j.sna.2011.04.001.

[2] B. P. Mann and B. A. Owens. “Investigations of a nonlinear energy harvester with a bistable potential well”. In: *Journal of Sound and Vibration* 329.9 (2010), pp. 1215–1226. ISSN: 0022460X. DOI: 10.1016/j.jsv.2009.11.034. URL: http://dx.doi.org/10.1016/j.jsv.2009.11.034.

[3] Enrico Dallago, Marco Marchesi, and Giuseppe Venchi. “Analytical model of a vibrating electromagnetic harvester considering nonlinear effects”. In: *IEEE Transactions on Power Electronics* 25.8 (2010), pp. 1989–1997. ISSN: 08858993. DOI: 10.1109/TPEL.2010.2044893.

[4] Abu Riduan Md Foisal, Chinsuk Hong, and Gwiy Sang Chung. “Multi-frequency electromagnetic energy harvester using a magnetic spring cantilever”. In: *Sensors and Actuators, A: Physical* 182 (2012), pp. 106–113. ISSN: 09244247. DOI: 10.1016/j.sna.2012.05.009. URL: http://dx.doi.org/10.1016/j.sna.2012.05.009.

[5] Marco P Soares dos Santos et al. “Magnetic levitation-based electromagnetic energy harvesting: a semi-analytical non-linear model for energy transduction”. In: *Scientific Reports* 6.January (2016), p. 18579. ISSN: 2045-2322. DOI: 10.1038/srep18579. URL: http://www.nature.com/articles/srep18579.

[6] D F Berdy, D J Valentino, and D Peroulis. “Sensors and Actuators A : Physical Design and optimization of a magnetically sprung block magnet vibration energy harvester”. In: *Sensors {&} Actuators: A. Physical* 218 (2014), pp. 69–79. ISSN: 0924-4247. DOI: 10.1016/j.sna.2014.06.011. URL: http://dx.doi.org/10.1016/j.sna.2014.06.011.

[7] N. V. Viet et al. “Energy harvesting from ocean waves by a floating energy harvester”. In: *Energy* 112 (2016), pp. 1219–1226. ISSN: 03605442. DOI: 10.1016/j.energy.2016.07.019. URL: http://dx.doi.org/10.1016/j.energy.2016.07.019.

[8] R Olaru, R GhercÇŐ, and C Petrescu. “Analysis and design of a vibration energy harvester using permanent magnets”. In: *Revue Roumaine des Sciences Techniques Serie Electrotechnique et Energetique* 59.2 (2014), pp. 131–140. ISSN: 00354066. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84903576656{\&}partnerID=40{\&}md5=3026d7d6c157163982aebbe305ccd4bd.

[9] C Martín Saravia, José M Ramírez, and Claudio D Gatti. “A hybrid numerical-analytical approach for modeling levitation based vibration energy harvesters”. In: *Sensors and Actuators, A: Physical* 257 (2017), pp. 20–29. ISSN: 09244247. DOI: 10.1016/j.sna.2017.01.023. URL: http://dx.doi.org/10.1016/j.sna.2017.01.023.

[10] Eugen Diaconescu. “The use of NARX neural networks to predict chaotic time series”. In: *WSEAS Transactions on Computer Research* 3.3 (2008), pp. 182–191. ISSN: 1991-8755. URL: http://dl.acm.org/citation.cfm?id=1466884.1466892.

[11] K.S. Narendra and K Parthasarathy. “Identification and control of dynamical systems using neural networks”. In: *IEEE Transactions on Neural Networks* 1.1 (1990), pp. 4–27. ISSN: 10459227. DOI: 10.1109/72.80202. URL: http://ieeexplore.ieee.org/document/80202/.

[12] Abdullah Nammari and Hamzeh Bardaweel. “Design enhancement and non-dimensional analysis of magnetically-levitated nonlinear vibration energy harvesters”. In: *Journal of Intelligent Material Systems and Structures* (2017), p. 1045389X1769859. ISSN: 1045-389X. DOI: 10.1177/1045389X17698592. URL: http://journals.sagepub.com/doi/10.1177/1045389X17698592.

[13] D F Berdy, D J Valentino, and D Peroulis. “Kinetic energy harvesting from human walking and running using a magnetic levitation energy harvester”. In: *Sensors and Actuators, A: Physical* 222 (2015), pp. 262–271. ISSN: 09244247. DOI: 10.1016/j.sna.2014.12.006. URL: http://dx.doi.org/10.1016/j.sna.2014.12.006.

[14] Wei Wang et al. “Magnetic-spring based energy harvesting from human motions: Design, modeling and experiments”. In: *Energy Conversion and Management* 132 (2017), pp. 189–197. ISSN: 01968904. DOI: 10.1016/j.enconman.2016.11.026. URL: http://dx.doi.org/10.1016/j.enconman.2016.11.026.

[15] Krzysztof Kecik et al. “Energy harvesting from a magnetic levitation system”. In: *International Journal of Non-Linear Mechanics* 94.January (2017), pp. 200–206. ISSN: 00207462. DOI: 10.1016/j.ijnonlinmec.2017.03.021. URL: http://dx.doi.org/10.1016/j.ijnonlinmec.2017.03.021.

[16] A G Avila Bernal and L E Linares García. “The modelling of an electromagnetic energy harvesting architecture”. In: *Applied Mathematical Modelling* 36.10 (2012), pp. 4728–4741. ISSN: 0307-904X. DOI: 10.1016/j.apm.2011.12.007. URL: http://dx.doi.org/10.1016/j.apm.2011.12.007.

[17] M L Morgado et al. “Mathematical modelling of cylindrical electromagnetic vibration energy harvesters”. In: *International Journal of Computer Mathematics* 7160.August (2014), pp. 1–9. ISSN: 0020-7160. DOI: 10.1080/00207160.2014.884715. URL: http://www.tandfonline.com/doi/abs/10.1080/00207160.2014.884715.

[18] M Geisler et al. “Human-motion energy harvester for autonomous body area sensors”. In: ().

[19] B. P. Mann and N. D. Sims. “Energy harvesting from the nonlinear oscillations of magnetic levitation”. In: *Journal of Sound and Vibration* 319.1-2 (2009), pp. 515–530. ISSN: 0022460X. DOI: 10.1016/j.jsv.2008.06.011.

[20] R. Ravaud and G. Lemarquand. “Comparison of the Coulombian and Ampe- Rian Current Models for Calculating the Magnetic Field Produced By Radially Mag- Netized Arc-Shaped Permanent Magnets”. In: *Progress In Electromagnetics Research* 95 (2009), pp. 309–327.

[21] Progress In. “Inverse Equivalent Surface Current Method With Hierarchical Higher Order Basis Func-”. In: 106.July (2010), pp. 377–394.

[22] Ahmed Munaz, Byung Chul Lee, and Gwiy Sang Chung. “A study of an electromagnetic energy harvester using multi-pole magnet”. In: *Sensors and Actuators, A: Physical* 201 (2013), pp. 134–140. ISSN: 09244247. DOI: 10.1016/j.sna.2013.07.003. URL: http://dx.doi.org/10.1016/j.sna.2013.07.003.

[23] R. Ravaud et al. “Cylindrical magnets and coils: Fields, forces, and inductances”. In: *IEEE Transactions on Magnetics* 46.9 (2010), pp. 3585–3590. ISSN: 00189464. DOI: 10.1109/TMAG.2010.2049026.

[24] K. L. Du and M. N.S. Swamy. “Neural networks in a softcomputing framework”. In: *Neural Networks in a Softcomputing Framework* 2 (2006), pp. 1–566. ISSN: 1098-6596. DOI: 10.1007/1-84628-303-5. arXiv: arXiv:1011.1669v3. URL: http://books.google.ru/books?id=bUNJAAAACAAJ{\%}5Cnhttp://ecee.colorado.edu/academics/schedules/ECEN5120.pdf.

[25] Velimir Ćirović, Dragan Aleksendrić, and Dušan Mladenović. “Braking torque control using recurrent neural networks”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 226.6 (2012), pp. 754–766. ISSN: 0954-4070. DOI: 10.1177/0954407011428720. URL: http://journals.sagepub.com/doi/10.1177/0954407011428720.

[26] Martin T. Hagan, Howard B. Demuth, and Orlando De Jesús. “An introduction to the use of neural networks in control systems”. In: *International Journal of Robust and Nonlinear Control* 12.11 (2002), pp. 959–985. ISSN: 10498923. DOI: 10.1002/rnc.727.

[27] Sayan Saha et al. “Identification of nonlinear systems from the knowledge around different operating conditions: A feed-forward multi-layer ANN based approach”. In: *2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing* (2012), pp. 413–418. DOI: 10.1109/PDGC.2012.6449856. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6449856.

[28] Krzysztof Kecik et al. “Energy harvesting from a magnetic levitation system”. In: *International Journal of Non-Linear Mechanics* March (2017), pp. 1–7. ISSN: 00207462. DOI: 10.1016/j.ijnonlinmec.2017.03.021. URL: http://linkinghub.elsevier.com/retrieve/pii/S002074621730241X.

[29] C. R. Saha et al. “Electromagnetic generator for harvesting energy from human motion”. In: *Sensors and Actuators, A: Physical* 147.1 (2008), pp. 248–253. ISSN: 09244247. DOI: 10.1016/j.sna.2008.03.008.

[30] Wenjia Zhao. “Design Optimization of a Linear Generator With \r\nDual Halbach Array for Human Motion Energy Harvesting”. In: *Theses, Dissertations, and Student Research from Electrical & Computer Engineering* (2015). URL: http://digitalcommons.unl.edu/elecengtheses/65.

[31] Dynamixel MX-106. *Mx-106T / Mx-106R*. 2010.

[32] Aida A. Ferreira, Teresa B. Ludermir, and Ronaldo R B De Aquino. “Comparing recurrent networks for time-series forecasting”. In: *Proceedings of the International Joint Conference on Neural Networks* May 2015 (2012). ISSN: 2161-4393. DOI: 10.1109/IJCNN.2012.6252459.

[33] Martin T. Hagan and Mohammad B. Menhaj. “Training Feedforward Networks with the Marquardt Algorithm”. In: *IEEE Transactions on Neural Networks* 5.6 (1994), pp. 989–993. ISSN: 19410093. DOI: 10.1109/72.329697.

[34] G E P Box, G M Jenkins, and G C Reinsel. “Time Series Analysis: Forecasting & Control”. In: *Book* January 2015 (1994), p. 709. ISSN: 1467-9280. DOI: 10.1016/j.ijforecast.2004.02.001. arXiv: arXiv:1011.1669v3.

[35] Greg Heath. *MATLAB Central - On Designing a Feedback Time-Series Neural Network...* URL: http://comp.soft-sys.matlab.narkive.com/rdtU8f2n/on-designing-a-feedback-time-series-neural-network-for-operational-deployment.

[36] R. Rohith Krishnan and S. Krishnakumar. “A Scheme for Analysis and Design of Analogue Circuits”. In: *Indian Journal of Science and Technology* 9.45 (2016). ISSN: 0974-5645. DOI: 10.17485/ijst/2016/v9i44/74155. URL: http://www.indjst.org/index.php/indjst/article/view/74155.

Appendices

Appendix A

Generator/Testing Station

A.1 Generator

A.1.1 Components and drawings

Article ID	ITN-25
Material	NdFeB
strength	approx. 14 kg
Pot diameter D	25 mm
Pot height H	7 mm
Thread M	5
Magnetisation	N38
Coating	Nickel (Ni)
Max. working temperature	80°C
Tolerance	+/- 0.1 mm
Steel	Q235 (China)
Weight	21 g

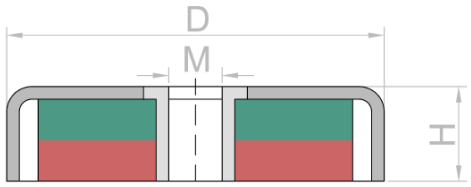


Figure A.1: Levitation array Magnets

Shape	Ring	
Outer diameter	26,75 mm	
Inner diameter	16 mm	
Height	5 mm	
Tolerance in size	+/- 0,1 mm	
Direction of magnetisation	axial (parallel to height)	
Material	NdFeB (Neodymium Iron Boron)	
Type of coating	Nickel (Ni-Cu-Ni)	
Strength	approx. 8,3 kg	approx. 81,4 N
Weight	13,7157 g	
Manufacturing method	sintered	
Magnetisation (Grade)	N42	
Max. working temperature	80°C	
Curie temperature	310 °C	
Residual magnetism Br	12900-13200 G	1.29-1.32 T
Coercive field strength bHc	10.8-12.0 kOe	860-955 kA/m
Coercive field strength iHc	≥12 kOe	≥955 kA/m
Energy product (BxH)max	40-42 MGOe	318-334 kJ/m^3

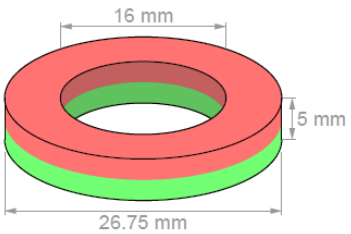
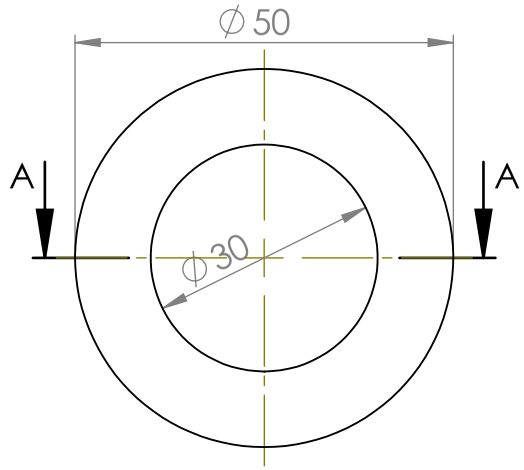
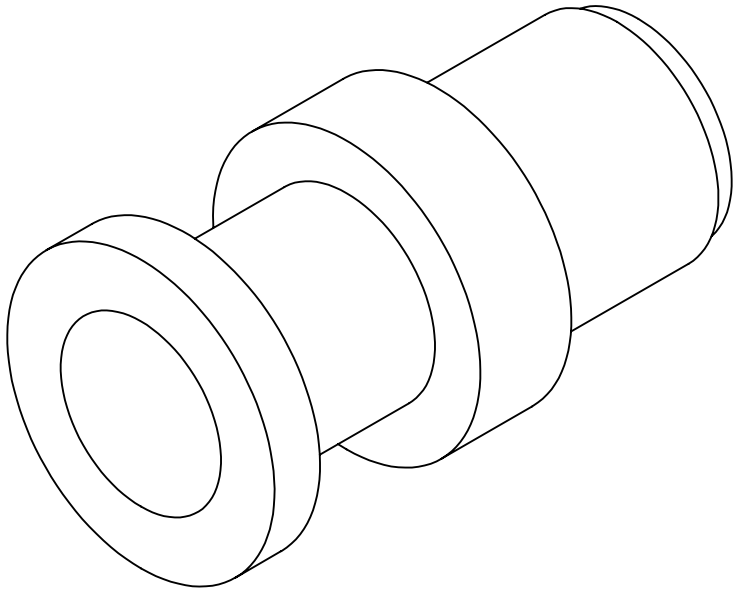
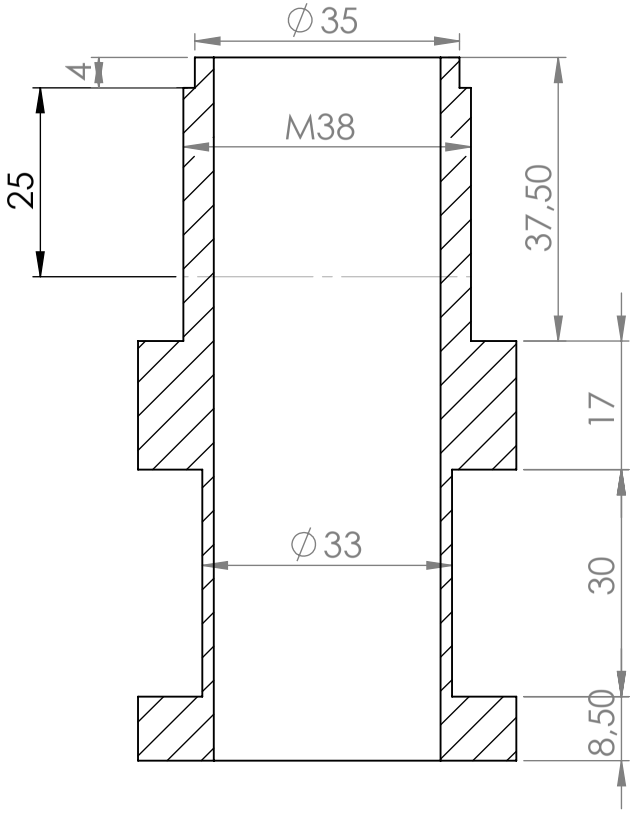


Figure A.2: Fixed Magnets

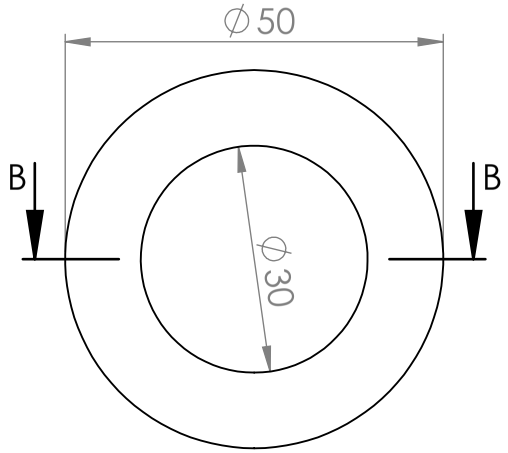


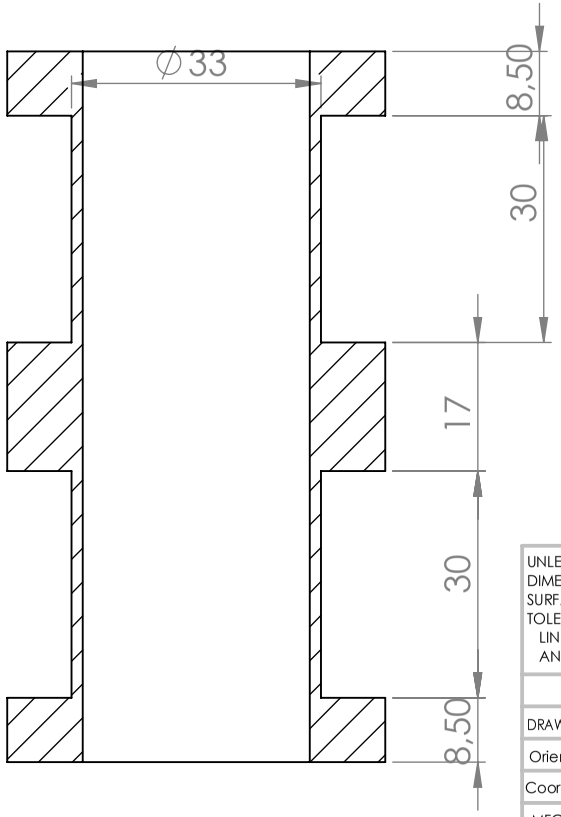




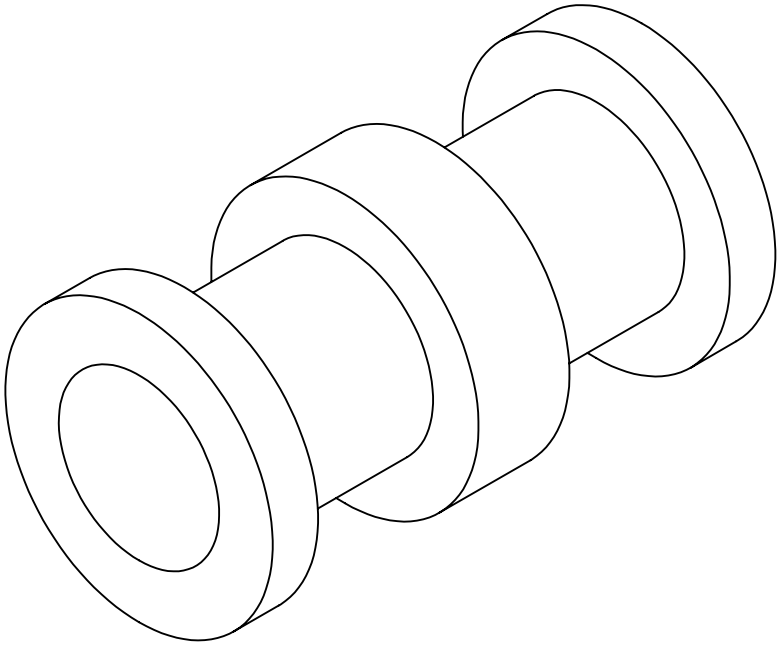
SECTION A-A
SCALE 1 : 1

UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					corpo_S1	
Orient.	Marco Santos						
Coorient.	Jorge Ferreira						
MFG							
Q.A				MATERIAL: Ertalon 6 SA		DWG NO.	A4
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1

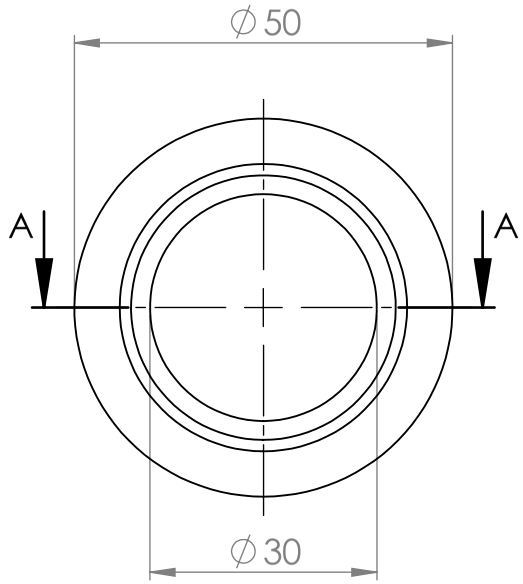


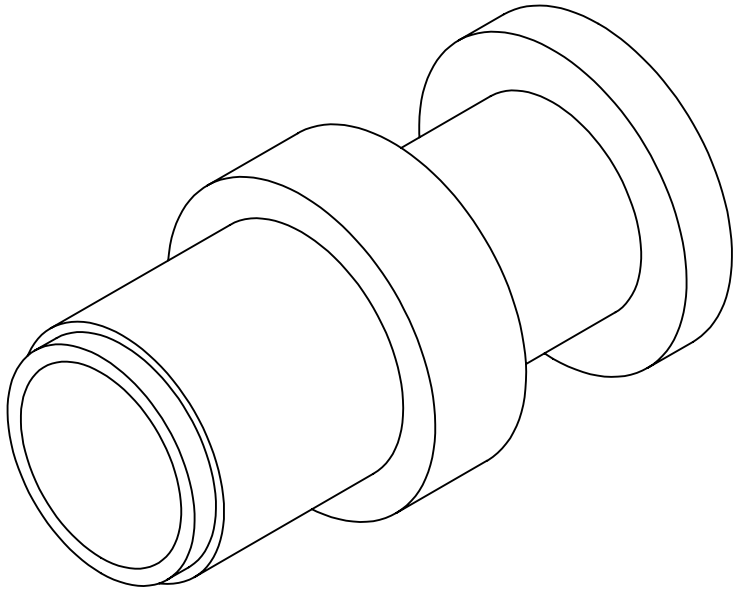


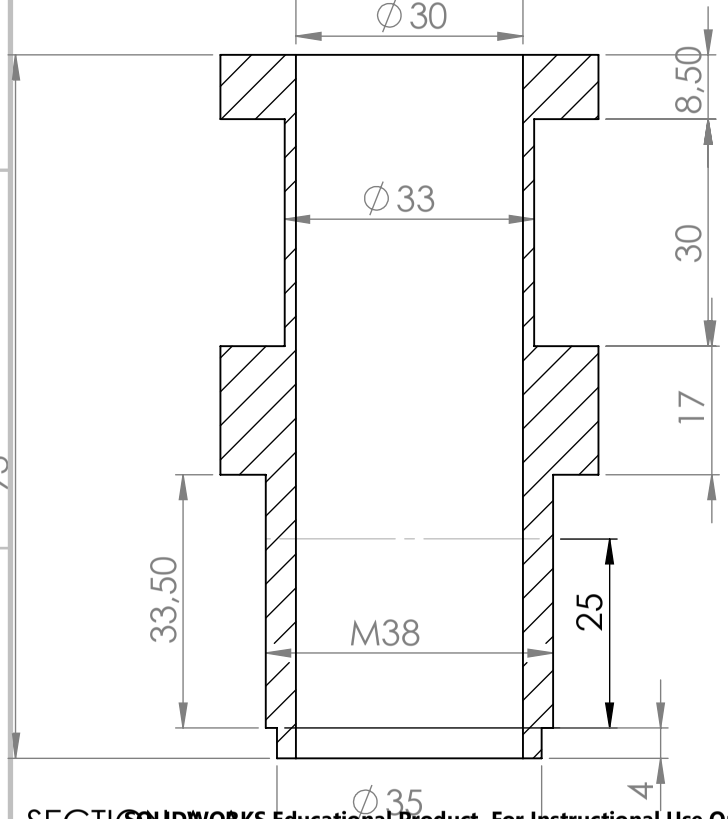
SECTION B-B
SCALE 1 : 1



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					Corpo_S2	
Orient.	Marco Santos						
Coorient.	Jorge Ferreira						
MFG							
Q.A				MATERIAL:		DWG NO.	A4
				Ertalon 6 SA			
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1

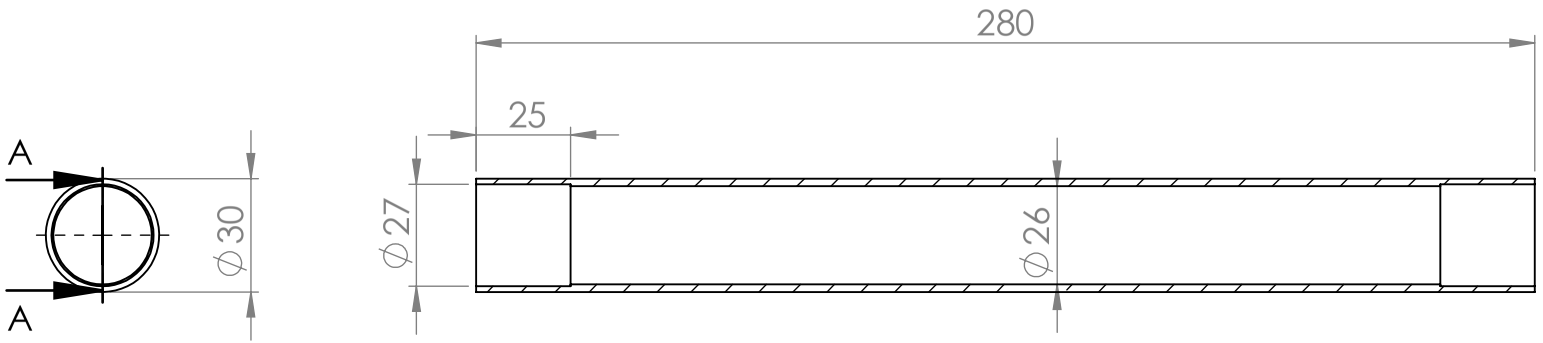






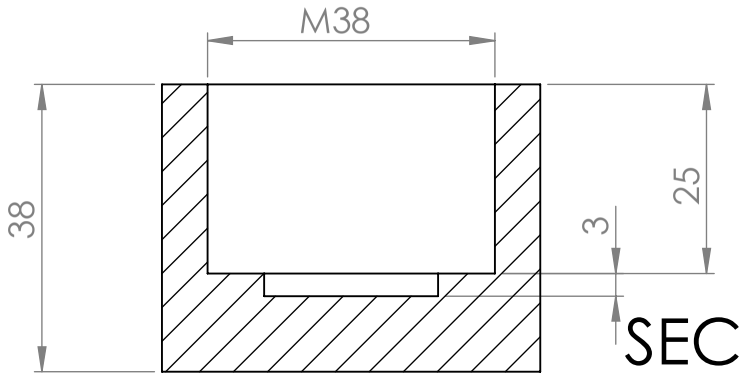
SECTION A-A
SCALE 1 : 1

UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					Corpo_s3	
Orient.	Marco Santos						
Coorient.	Jorge Ferreira						
MFG							
Q.A				MATERIAL:		DWG NO.	A4
				Ertalon 6SA			
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1

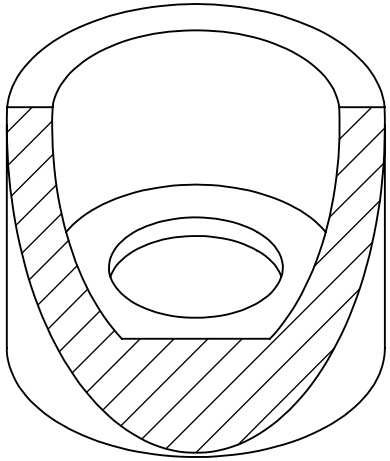


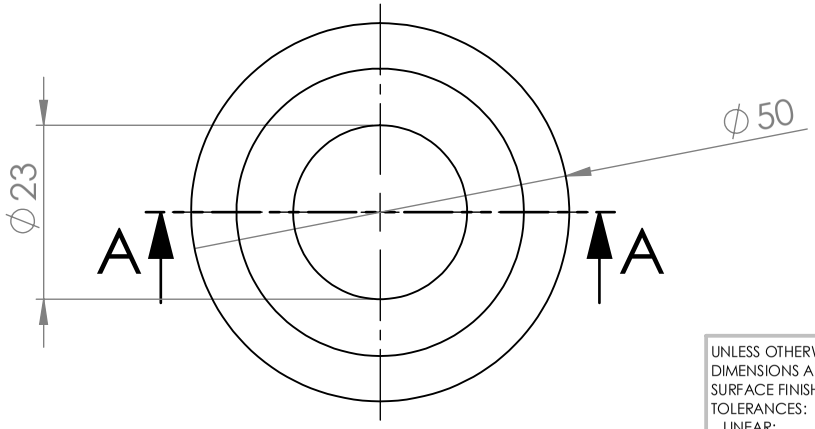
SECTION A-A
SCALE 1 : 2

UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:	FINISH:				DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING / REVISION
	NAME	SIGNATURE	DATE			TITLE: tubo acrílico
DRAWN	Renato Rocha					
Orient.	Marco Santos					
Coorient.	Jorge Ferreira					
MFG						
Q.A				MATERIAL:		DWG NO. / A4
				WEIGHT:		SCALE:1:2 / SHEET 1 OF 1

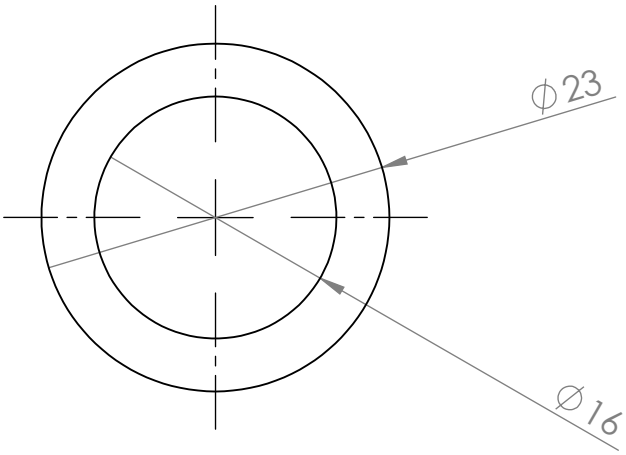


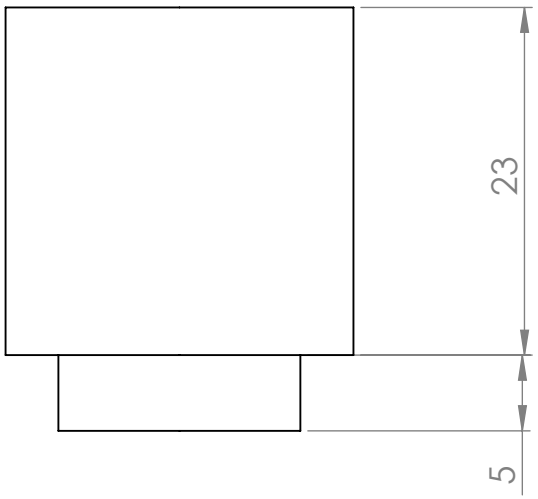
SECTION A-A



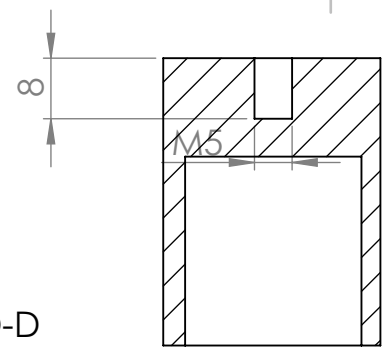


UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE: tampa_ext_nh	
DRAWN	Renato Rocha						
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1

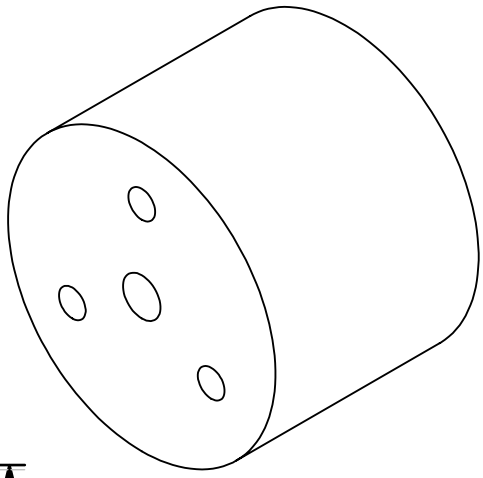


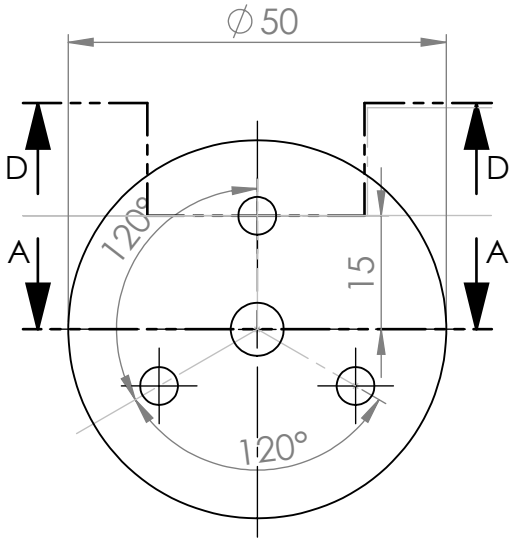


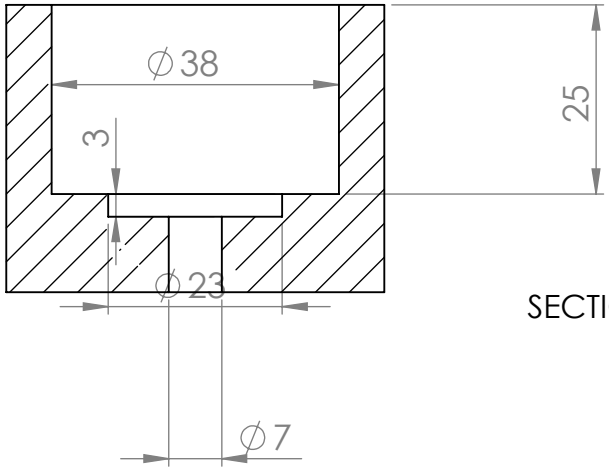
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					tampa_int_nh	
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
SOLIDWORKS Educational Product. For Instructional Use Only							
				WEIGHT:		SCALE:2:1	SHEET 1 OF 1



SECTION D-D

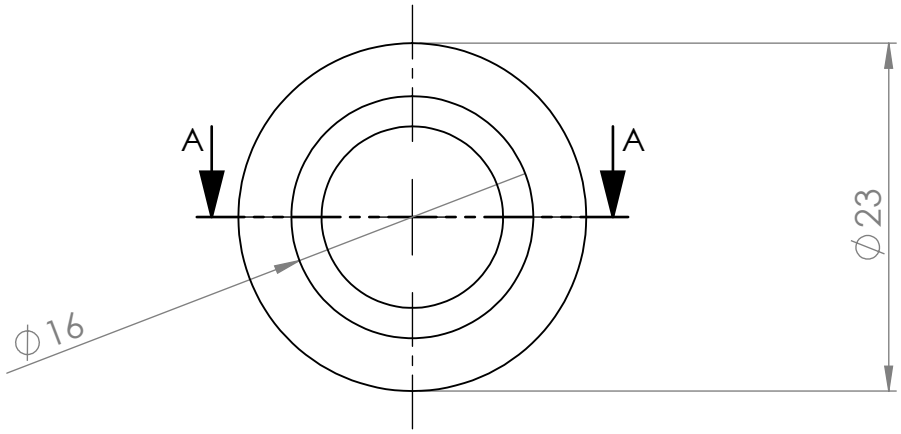


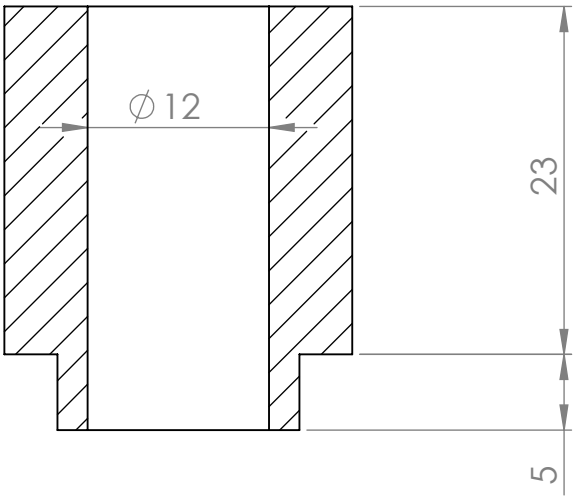




SECTION A-A

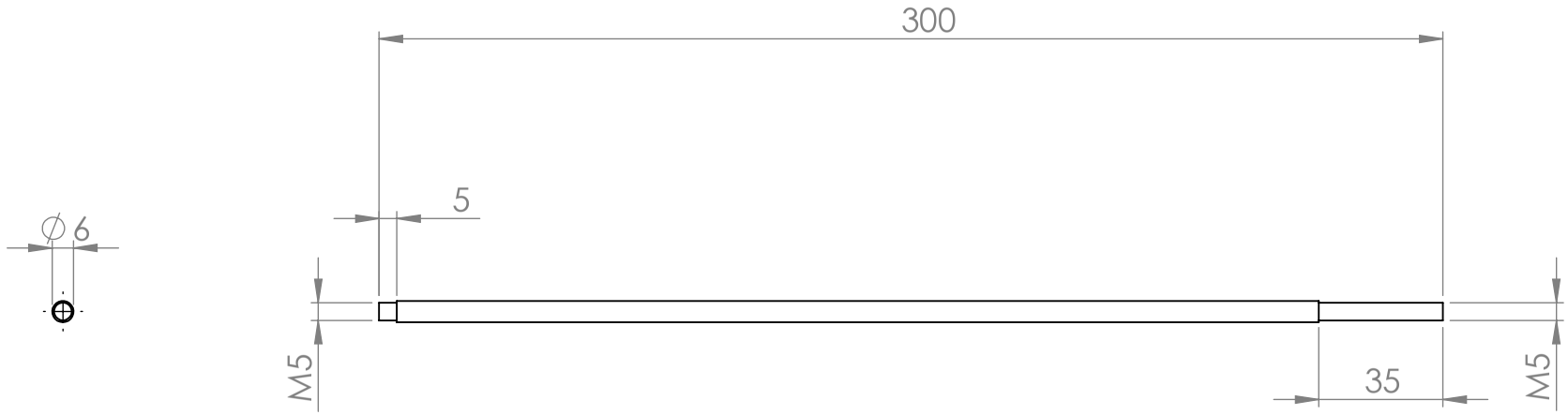
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					Tampa_ext_wh	
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
SOLIDWORKS Educational Product. For Instructional Use Only							
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1



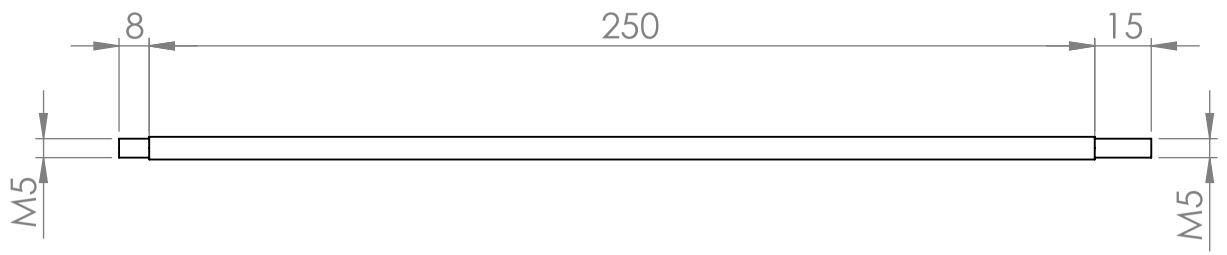


SECTION A-A

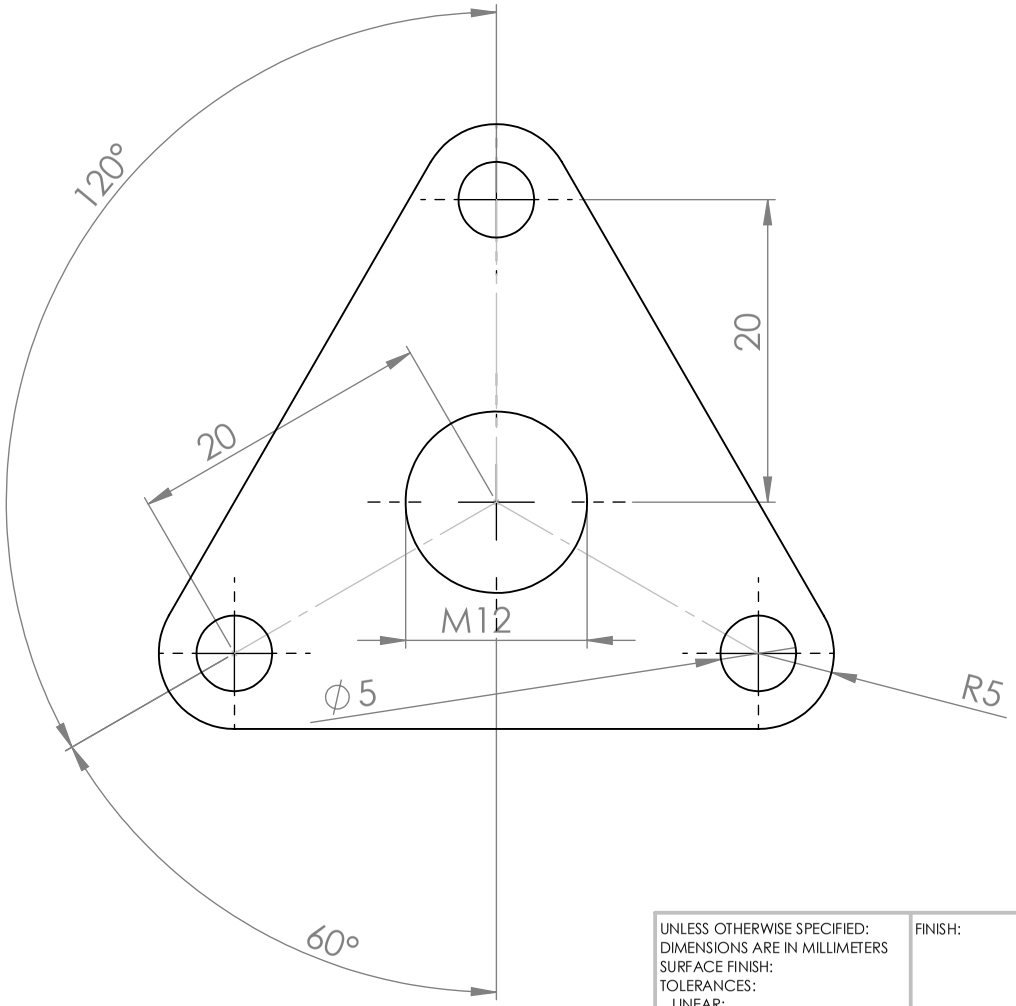
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					tampa_int_wh	
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
				Polímero			
				WEIGHT:		SCALE:2:1	SHEET 1 OF 1

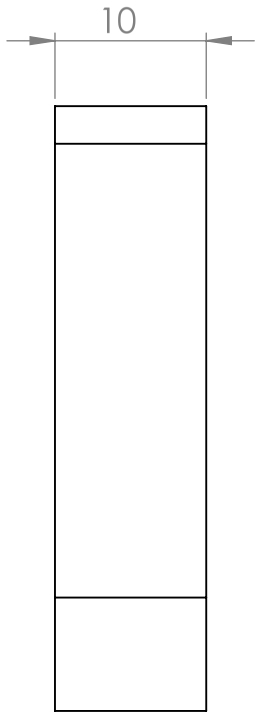


UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:	FINISH:				DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING / REVISION
	NAME	SIGNATURE	DATE			TITLE: Veio Levitação
DRAWN	Renato Rocha					
Orient.	Marco Santos					
Coorient.	Jorge Ferreira					
MFG						
Q.A				MATERIAL:		DWG NO. A4
				WEIGHT:		SCALE:1:2 / SHEET 1 OF 1

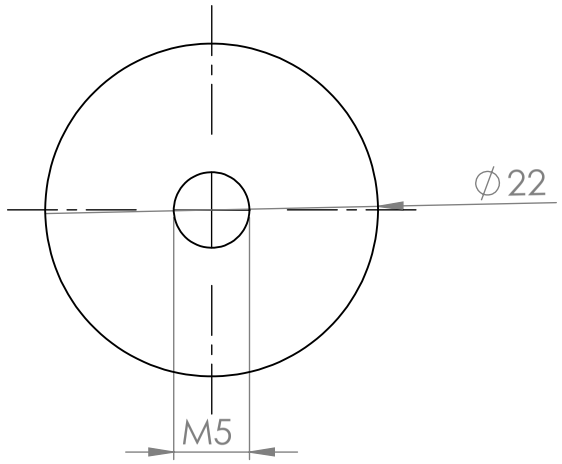


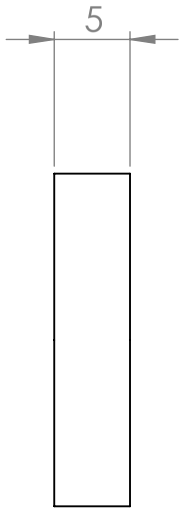
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:			DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING / REVISION
	NAME	SIGNATURE	DATE			TITLE: veio_suporte(x3)
DRAWN	Renato Rocha					
Orient.	Marco Santos					
Coorient.	Jorge Ferreira					
MFG						
Q.A				MATERIAL:		DWG NO. / A4
				WEIGHT:		SCALE:1:2 / SHEET 1 OF 1





UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:	FINISH:		DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION	
	NAME	SIGNATURE	DATE	TITLE: Suporte Sensor		
DRAWN	Renato Rocha					
Orient.	Marco Santos					
Coorient.	Jorge Ferreira					
MFG						
Q.A				MATERIAL:	DWG NO.	A4
				WEIGHT:	SCALE:2:1	SHEET 1 OF 1





UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:	FINISH:			DEBURR AND BREAK SHARP EDGES	DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE		TITLE:	
DRAWN	Renato Rocha				Prato	
Orient.	Marco Santos					
Coorient.	Jorge Ferreira					
MFG						
Q.A				MATERIAL:	DWG NO.	A4
				WEIGHT:	SCALE:2:1	SHEET 1 OF 1

A.1.2 Output Computing

```
%Variable assignment
R1=788.2;
Rc=785;
Rt=R1+Rc;
%Load acquired values
load('all_coils_a20_p2.mat')
load('all_coils_a25_p2.mat')
%Test 1 values
coil1=10*all_coils_a20_p2.Y(1).Data;
coil2=10*all_coils_a20_p2.Y(2).Data;
coil3=10*all_coils_a20_p2.Y(3).Data;
coil4=10*all_coils_a20_p2.Y(4).Data;
coil1=sqrt(coil1.^2);
coil2=sqrt(coil2.^2);
coil3=sqrt(coil3.^2);
coil4=sqrt(coil4.^2);

%Test 2 values
coil11=10*all_coils_a25_p2.Y(1).Data;
coil21=10*all_coils_a25_p2.Y(2).Data;
coil31=10*all_coils_a25_p2.Y(3).Data;
coil41=10*all_coils_a25_p2.Y(4).Data;
coil11=sqrt(coil11.^2);
coil21=sqrt(coil21.^2);
coil31=sqrt(coil31.^2);
coil41=sqrt(coil41.^2);
%Coil output summation
V={};
V{1,1}=coil1+coil2+coil3+coil4;
V{2,1}=coil11+coil21+coil31+coil41;
V{1,1}=V{1,1}(1,200:end-600);
V{2,1}=V{2,1}(1,200:end-600);
%Power computing cycle
for i=1:1
    for j=1:2
        t{j,i}=linspace(0,10,size(V{j,i},2)); %time vector
        I{j,i}=V{j,i}/Rt; %Current calculation
        P{j,i}=I{j,i}.*V{j,i}; %Power calculation
        MaxV{j,i}=max(V{j,i}); %Max voltage
        idx=find(V{j,i}==MaxV{j,i}); %Max voltage index
        MaxP{j,i}=MaxV{j,i}*I{j,i}(1,idx); %Max power
        avgP{j,i}=trapz(t{j,i},P{j,i})/10; %Average Power
        E{j,i}=trapz(t{j,i},P{j,i}); %Energy= Power intergal
    end
end
```

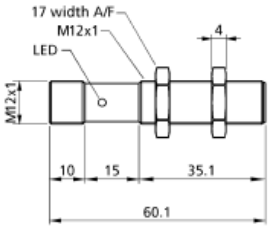
Figure A.3: Generator Outputs computing script

A.2 Testing station

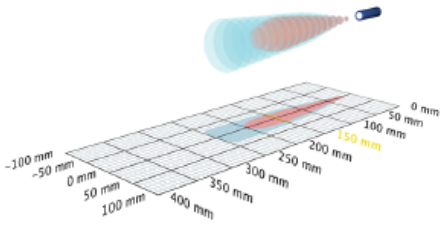
A.2.1 Components and drawings

nano-15/CU

scale drawing



detection zone



1 x analogue 0-10 V

250 mm

operating range	20 - 250 mm
design	cylindrical M12
operating mode	analogue distance measurements
particularities	narrow sound field

ultrasonic-specific

means of measurement	echo propagation time measurement
transducer frequency	380 kHz
blind zone	20 mm
operating range	150 mm
maximum range	250 mm
reproducibility	± 0.15 %
accuracy	± 1 % (temperature drift internally compensated)

electrical data

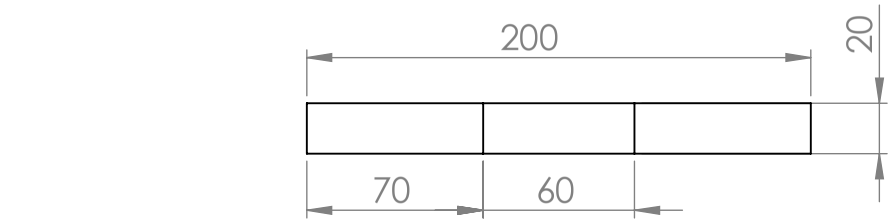
operating voltage U_B	15 V bis 30 V DC, verpolfest
voltage ripple	± 10 %
no-load current consumption	≤ 25 mA
type of connection	4-pin M12 initiator plug

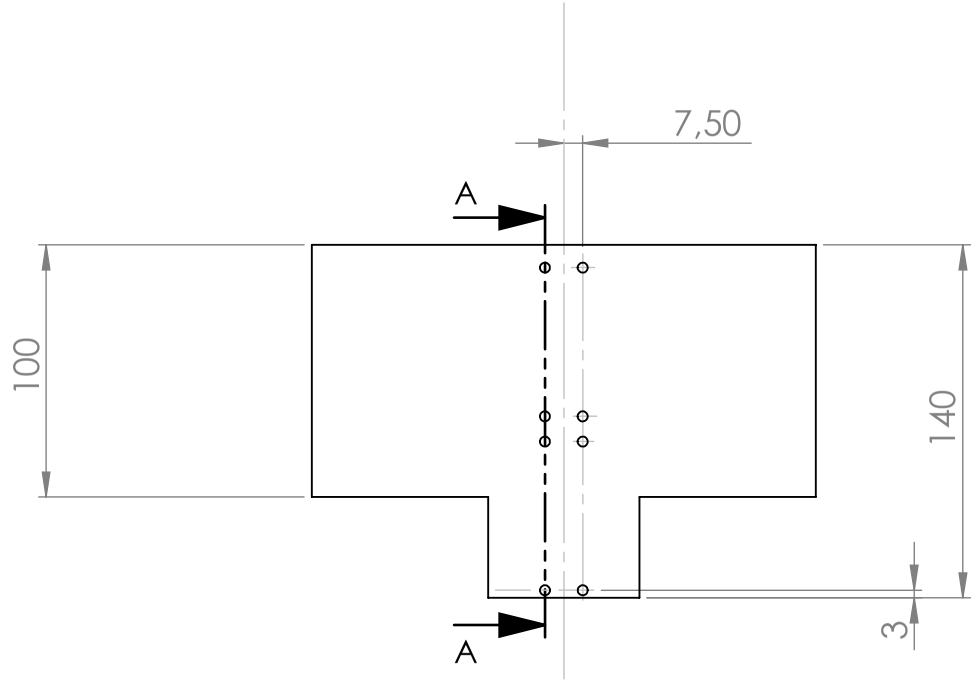
Figure A.4: Sensor data sheet 1

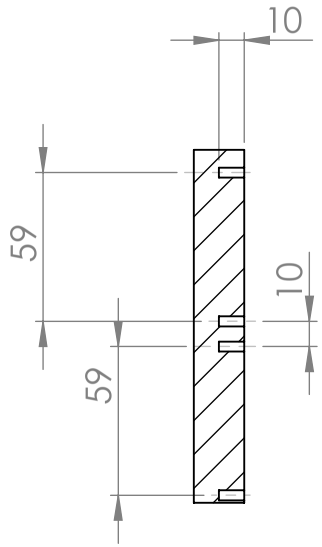
nano-15/CU

outputs	
output 1	analogue output voltage: 0-10 V (at $U_B \geq 15$ V), short-circuit-proof switchable rising/falling
response time	24 ms
delay prior to availability	< 300 ms
inputs	
input 1	Teach-in input
housing	
material	brass sleeve, nickel-plated, plastic parts, PBT
ultrasonic transducer	polyurethane foam, epoxy resin with glass contents
class of protection to EN 60529	IP 67
operating temperature	-25°C to +70°C
storage temperature	-40°C to +85°C
weight	15 g
technical features/characteristics	
scope for settings	Teach-in Teach-in über Com-Eingang an Pin 2
indicators	1 x LED green: working, 1 x LED yellow: object in the window
particularities	narrow sound field

Figure A.5: Sensor data sheet 2

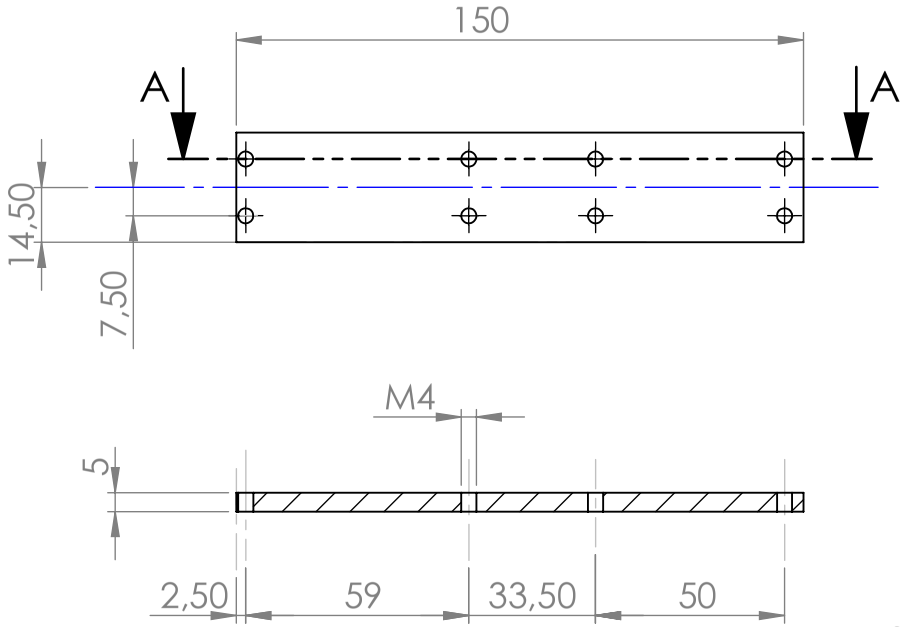






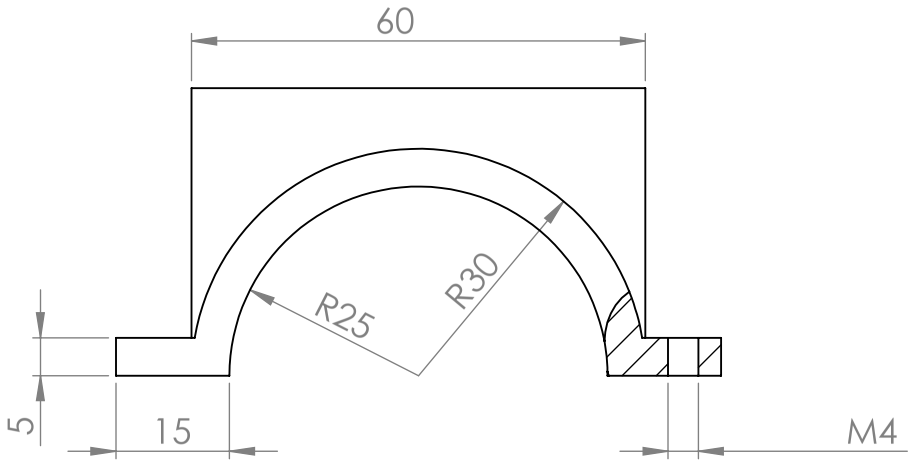
SECTION A-A

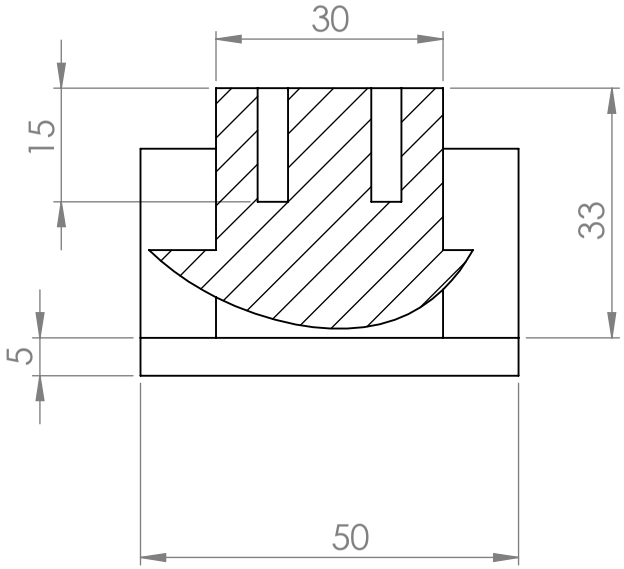
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					Base	
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
				Alumínio			
				WEIGHT:		SCALE:1:3	SHEET 1 OF 1

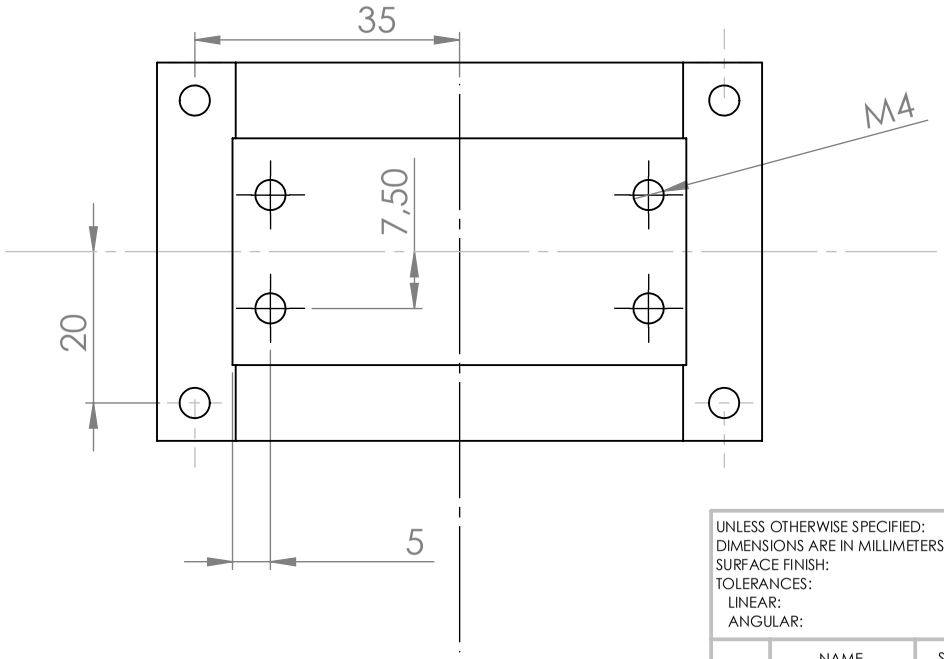


SECTION A-A

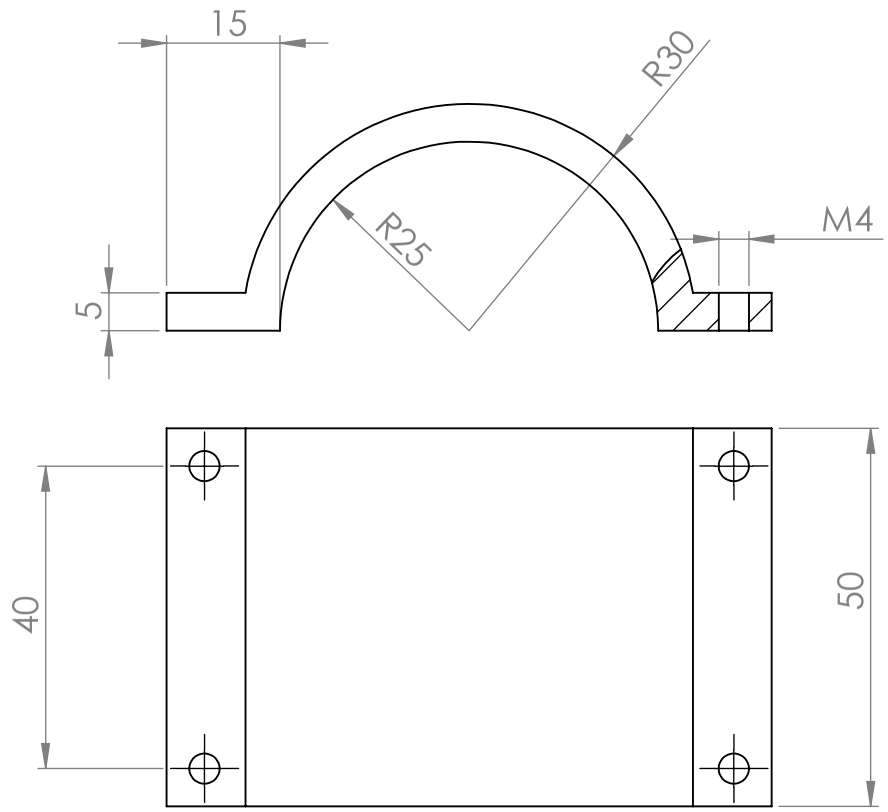
UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE: Suporte_gerador	
DRAWN	Renato Rocha						
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL: Alumínio		DWG NO.	A4
				WEIGHT:		SCALE:1:2	SHEET 1 OF 1







UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE: Grip_cima_drawing	
DRAWN	Renato Rocha						
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL: Acrílico		DWG NO.	A4
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1



UNLESS OTHERWISE SPECIFIED: DIMENSIONS ARE IN MILLIMETERS SURFACE FINISH: TOLERANCES: LINEAR: ANGULAR:		FINISH:		DEBURR AND BREAK SHARP EDGES		DO NOT SCALE DRAWING	REVISION
	NAME	SIGNATURE	DATE			TITLE:	
DRAWN	Renato Rocha					Grip_inferior_drawing	
Orient.	Jorge Ferreira						
Coorient.	Marco Santos						
				MATERIAL:		DWG NO.	A4
				Acrílico			
				WEIGHT:		SCALE:1:1	SHEET 1 OF 1

A.2.2 Code excerpts

```
void CheckSum(byte dat[], int d)
{

  int c;
  int i;
  int idx;
  c = 0;
  i = 0;
  idx = 0;
  for (idx = 2; idx < d; idx++)
  {
    c += dat[idx];
  }
  i = c & 0xFF;

  i = ~i;
  dat[d] = i;
}
```

Figure A.6: Function to automatically generate CheckSum byte

```
void AngToReg(float ang, byte dat[])
{
  float val;
  int intval;
  val = ang * (4095.0 / 360);
  intval = int(val);
  byte reg1 = intval & 0xff;
  byte reg2 = (intval >> 8) & 0xff;

  dat[6] = reg1;
  dat[7] = reg2;

}
```

Figure A.7: Function to automatically allocate angle value into correct memory registers

Appendix B

NARX implementation Codes and Simulation Diagrams

B.1 NARX implementation and training codes

```
% NARX designing, training and testing

%File Reading
[WaveAngles, Mag_Position]=Read_File('final_data.txt');

inputSeries = WaveAngles;
targetSeries = Mag_Position;


% NARX implementation and parameters definition
inputDelays = 1:7;
feedbackDelays = 1:7;
hiddenLayerSize = 10;

net = narxnet(inputDelays,feedbackDelays,hiddenLayerSize);

% Data Preparation and division
[inputs,inputStates,layerStates,targets] = preparets(net,inputSeries,{},targetSeries);

net.divideFcn='divideblock';
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Network training and previous learning algorithm selection

%net.trainFcn = 'trainlm'; %Levenberg-Marquardt
%net.trainFcn = 'trainbr'; %Bayesian Regulation
%net.trainFcn = 'trainscg'; %Scaled Conjugate Gradient

net.trainParam.epochs=1500;
[net,tr] = train(net,inputs,targets,inputStates,layerStates);

% Test the Network
outputs = net(inputs,inputStates,layerStates);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs);
```

Figure B.1: Network implementation and training using the NN toolbox functions

```
netc = closeloop(net);
[inputs,inputStates,layerStates,targets] = preparets(netc,inputSeries,{},targetSeries);
[netc,trc]=train(netc,inputs,targets,inputStates,layerStates);

performancec = perform(netc,targets,outputs);
```

Figure B.2: Network code excerpt for close loop implementation and training

B.2 Simulation Diagrams

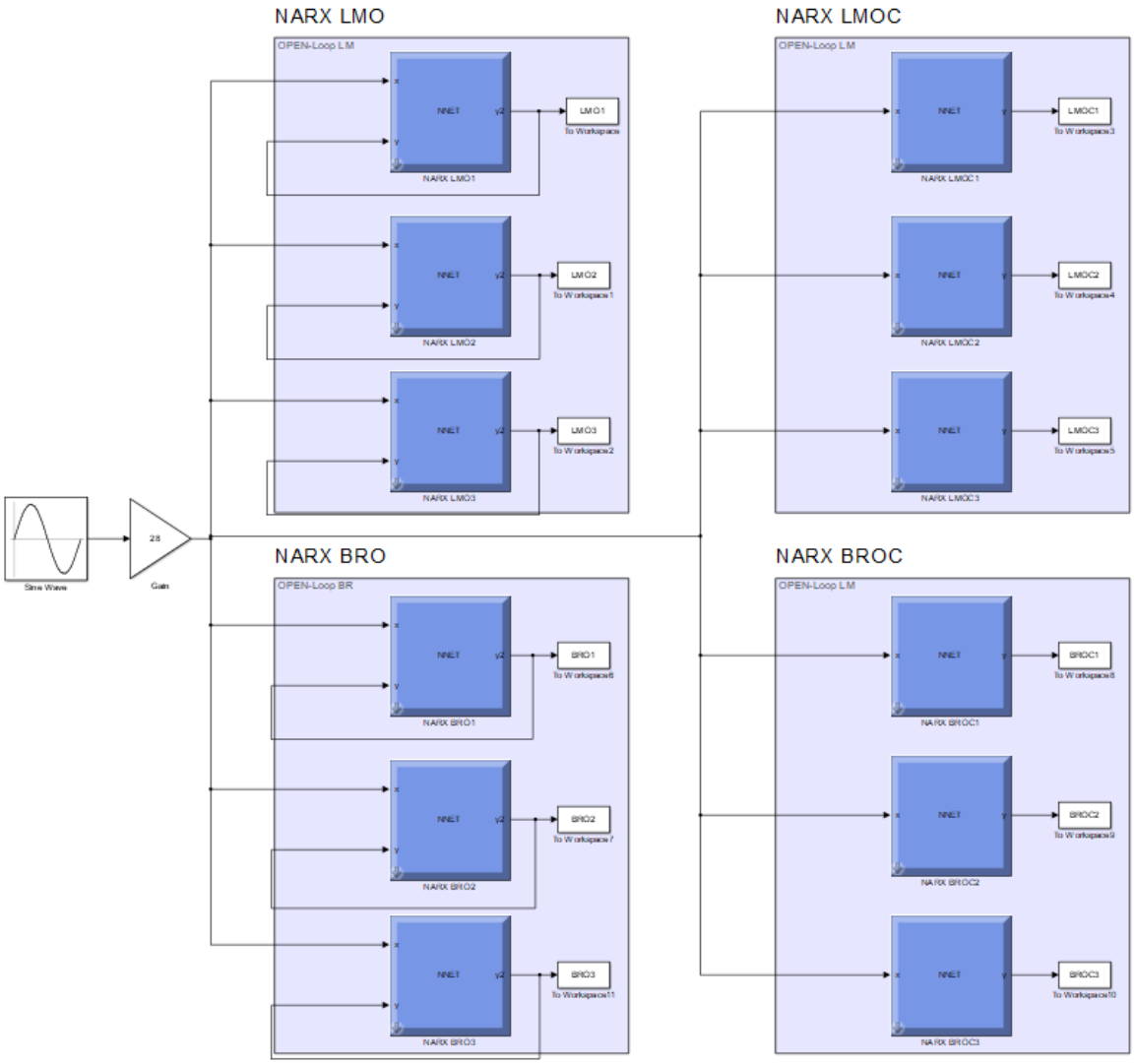


Figure B.3: NARX training configurations simulation diagram

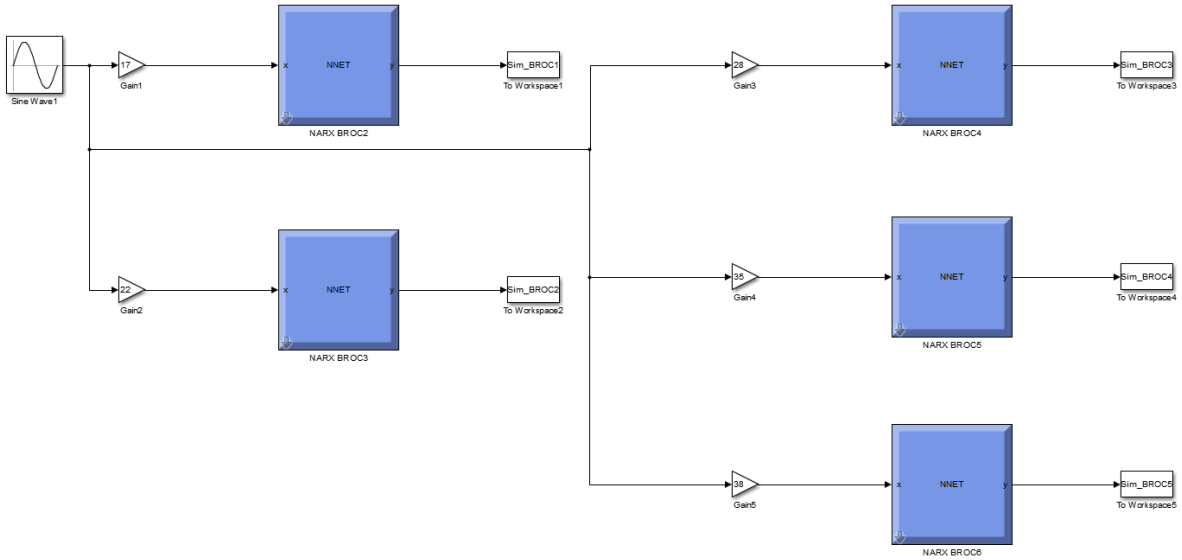


Figure B.4: NARX BROC simulation diagram

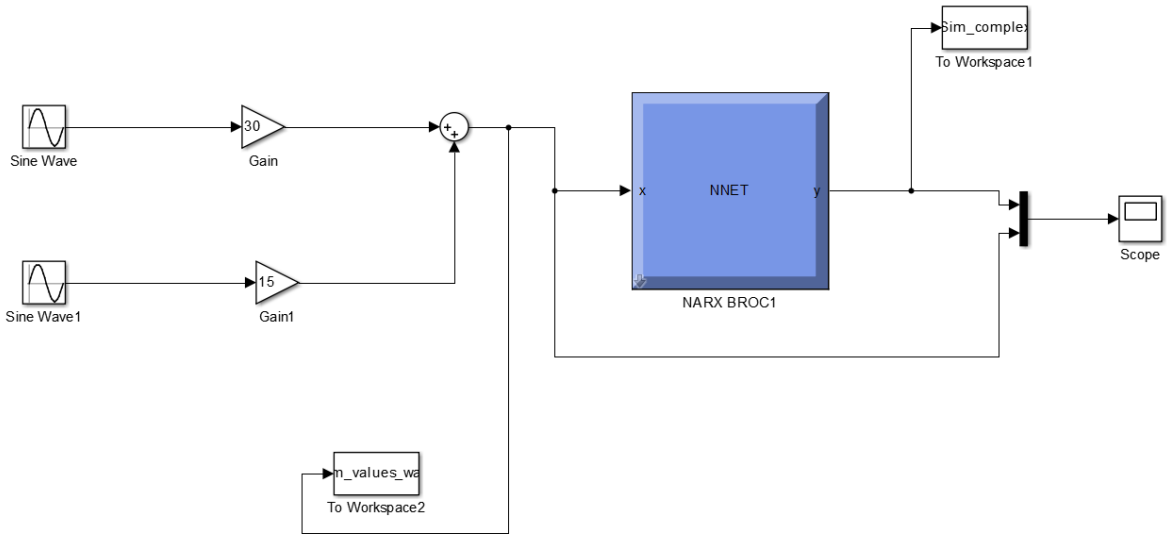


Figure B.5: NARX BROC complex wave simulation diagram

B.3 Error computing

```
[ExpWaveAng, ExpMag_Pos]=Read_File('test_P1,5_A28.txt'); %Experimental values reading
ExpMag_Pos=cell2mat(ExpMag_Pos); %Convertion of cell to double

%Resample of experimental vector into same size as simulation vector
ExpMag_Pos_ts=resample(ExpMag_Pos,size(BRO_mat,2),size(ExpMag_Pos,2));
%MSE computation
for i=1:3
mseBRO(1,i) = immse(ExpMag_Pos_ts, BRO_mat(i,:));
mseBROC(1,i) = immse(ExpMag_Pos_ts, BROC_mat(i,:));
mseLMO(1,i) = immse(ExpMag_Pos_ts, LMO_mat(i,:));
mseLMOC(1,i) = immse(ExpMag_Pos_ts, LMOC_mat(i,:));
end
%MPE computation
for k=1:3
MPE_BRO(1,k) =(100/size(BRO_mat(k,:),2))*sum((ExpMag_Pos_ts-BRO_mat(k,:))./ExpMag_Pos_ts) ;
MPE_BROC(1,k) =(100/size(BROC_mat(k,:),2))*sum((ExpMag_Pos_ts-BROC_mat(k,:))./ExpMag_Pos_ts) ;
MPE_LMO(1,k) = (100/size(LMO_mat(k,:),2))*sum((ExpMag_Pos_ts-LMO_mat(k,:))./ExpMag_Pos_ts );
MPE_LMOC(1,k) = (100/size(LMOC_mat(k,:),2))*sum((ExpMag_Pos_ts-LMOC_mat(k,:))./ExpMag_Pos_ts );
end
%Cross correlation coefficient computation
for c=1:3
CC_BRO(1,c) = xcorr(BRO_mat(c,:),ExpMag_Pos_ts,0,'coeff');
CC_BROC(1,c) = xcorr(BROC_mat(c,:),ExpMag_Pos_ts,0,'coeff');
CC_LMO(1,c) = xcorr(LMO_mat(c,:),ExpMag_Pos_ts,0,'coeff');
CC_LMOC(1,c) = xcorr(LMOC_mat(c,:),ExpMag_Pos_ts,0,'coeff');
end
```

Figure B.6: Error Computing Script pt.1

```
%Average CC
avgCC_BRO=mean(CC_BRO);
avgCC_BROC=mean(CC_BROC);
avgCC_LMO=mean(CC_LMO);
avgCC_LMOC=mean(CC_LMOC);

%Maximum CC
maxCC_BRO=max(CC_BRO);
maxCC_BROC=max(CC_BROC);
maxCC_LMO=max(CC_LMO);
maxCC_LMOC=max(CC_LMOC);

%Average MSE
avgmseBRO=mean(errBRO);
avgmseBROC=mean(errBROC);
avgmseLMO=mean(errLMO);
avgmseLMOC=mean(errLMOC);

%Average MSE
minmseBRO=min(errBRO);
minmseBROC=min(errBROC);
minmseLMO=min(errLMO);
minmseLMOC=min(errLMOC);

%Average MPE
avgMPEBRO=mean(abs(MPE_BRO));
avgMPEBROC=mean(abs(MPE_BROC));
avgMPELMO=mean(abs(MPE_LMO));
avgMPELMOC=mean(abs(MPE_LMOC));

%Minimum MPE
minMPEBRO=min(abs(MPE_BRO));
minMPEBROC=min(abs(MPE_BROC));
minMPELMO=min(abs(MPE_LMO));
minMPELMOC=min(abs(MPE_LMOC));
```

Figure B.7: Error Computing Script pt.2