MDPI

Article

Dynamics and Control of a Magnetic Transducer Array Using Multi-Physics Models and Artificial Neural Networks

Georgios Tsakyridis [1] and Nikolaos I. Xiros [2,*]

[1] School of Electrical and Computer Engineering, Aristotle University of Thessaloniki, University Campus, 54124 Thessaloniki, Greece; tsakyridis@ece.auth.gr
[2] School of Naval Architecture and Marine Engineering, University of New Orleans, New Orleans, LA 70148, USA
[*] Correspondence: nxiros@uno.edu

Abstract: A linear mechanical oscillator is non-linearly coupled with an electromagnet and its driving circuit through a magnetic field. The resulting non-linear dynamics are investigated using magnetic circuit approximations without major loss of accuracy and in the interest of brevity. Different computational approaches to simulate the setup in terms of dynamical system response and design parameters optimization are pursued. A current source operating in baseband without modulation directly feeds the electromagnet, which consists commonly of a solenoid and a horseshoe-shaped core. The electromagnet is then magnetically coupled to a mass made of soft magnetic material and attached to a spring with damping. The non-linear system is described by a linearized steady-space representation while is examined for controllability and observability. A controller using a pole placement approach is built to stabilize the element. Drawing upon the fact that coupling works both ways, enabling estimation of the mass position and velocity (state variables) by processing the induced voltage across the electromagnet, a state observer is constructed. Accurate and fast tracking of the state variables, along with the possibility of driving more than one module from the same source using modulation, proves the applicability of the electro-magneto-mechanical transducer for sensor applications. Next, a three-layer feed-forward artificial neural network (ANN) system equivalent was trained using the non-linear plant-linear controller-linear observer configuration. Simulations to investigate the robustness of the system with respect to different equilibrium points and input currents were carried out. The ANN proved robust with respect to position accuracy.

Keywords: sensor; magnetic; transducer; control; observer; multi-physics; nonlinear; dynamics; modelling; artificial neural network

1. Introduction

Advancements in the field of magnetic materials in terms of better efficiency and energy densities [1], together with the integration of mechanics, electromagnetics, power and control electronics into the system, have enabled intelligent electromagnetic actuators and sensors. Electromechanical devices are fundamentally based on quasi-static magnetic and electric fields, where force is generated [2]. Widely spread nowadays due to their simple design and system integration, they can measure changes in the magnetic reluctance of the circuit, which can be correspondingly related to a change in speed or position [3]. Precise operation can be realized using control techniques such as feedback control, which enables them both as sensors and actuators [4]. Typical applications can be found in aerospace [5], automotive [6], ocean [7] and biomedical devices [8] as well as flywheel and transmission speed [9], crank and cam shaft position for engine timing, throttle valve position for air intake, steering wheel position, pedal position, fluid level, chassis height, and in electronic door locks [6]. Given the joint sensing-actuation functionality, electro-magneto-mechanical devices are a promising solution for multi-agent architectures. Such architectures have a vast application field, such as mechatronics and robotics [10], smart grids [11] as well

as sensor networks [12]. An agent is a software-controlled hardware, which reacts to a stimulus according to a user-defined behavior [13–16]. A multi-agent constellation or swarm uses a collaborative approach to achieve a common purpose [17–23], difficult or impossible to solve by a monolithic system. In this work, we examine a sensor application of an electromechanical oscillator as proposed by Xiros [24,25] (Figure 1). Initially, a force imposed on the mechanical system by the electromagnet induces a displacement of the mass, altering the air gap length. In the next instance, the acting force has changed and the mass moves again. This iterative sequence repeats, forming a closed loop between the displacement and the magnetic force. The magnetically actuated mass-spring-damper system is described and simulated in Section 2. Different simulation methods are presented and compared. Section 3 shows the state-space formulation of the problem together with its linearized form. Using the eigenvalues concept, the system's' stability is assessed. In Section 4, closed-loop simulations are conducted to achieve a higher degree of stability. In order to estimate the position and velocity of the oscillating mass, the system states must be known at all times. By designing a state observer for the electromechanical oscillator, position and velocity tracking is achieved. Last, the overall system is replaced by an artificial neural network (ANN) equivalent. The ANN is trained using data collected from the analytical model for different equilibrium points. The current input is kept variable yet close to the equilibrium values where the linearization is still valid.
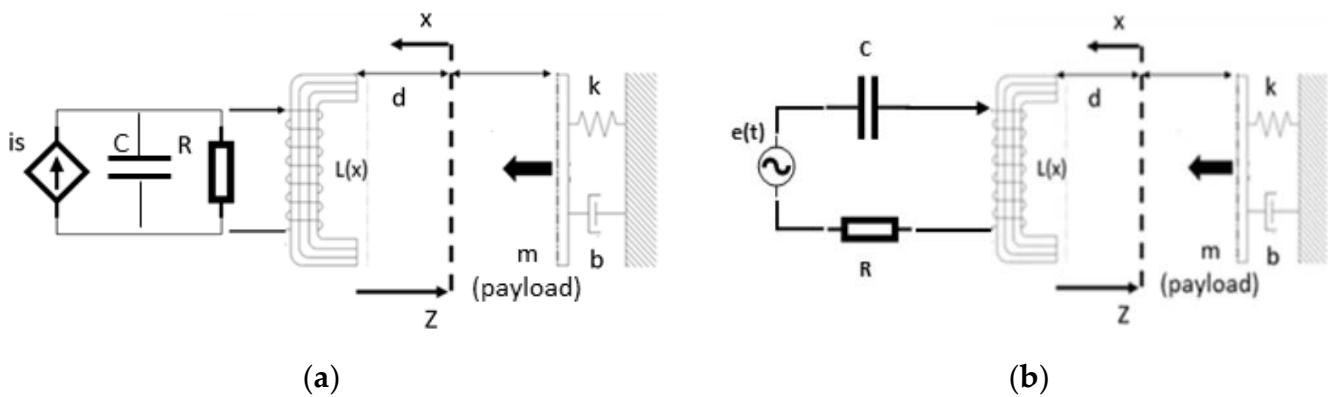

(**a**)   (**b**)

Figure 1. (**a**) Coupled electromechanical oscillator investigated in this work; (**b**) coupled electromechanical oscillator as described in [25].

2. The Electromechanical Oscillator

2.1. Lagrangian Formulation of the System

Figure 1a,b show the electromechanical oscillator investigated in this work and in [25] respectively.

Analysis of dynamic systems requires the equations of motion for such systems to be derived. The ability of computation and analytical approaches depends on the mathematical description used to formulate system dynamics. Often Newtonian mechanics is replaced by Lagrangian dynamics, which solely depends on the energy balance within a non-dissipative system:

$$\mathcal{L} = \mathbb{K} - \mathbb{T} \tag{1}$$

Lagrange's equation can be augmented, using the so-called "power" function, to include dissipative forces as well as other non-dissipative, non-conservative external forces:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = \frac{\partial P}{\partial \dot{q}_i} \tag{2}$$

Let us now consider the electromechanical system shown in Figure 1a. In specific, the system's circuit consists of an electric current source, a conductance, a capacitor and the

electromagnet's coil all connected in parallel. The Lagrangian of the system can then be written as:

$$\mathcal{L} = \mathbb{K}_e + \mathbb{K}_m - (\mathbb{T}_e + \mathbb{T}_m), \tag{3}$$

$$\mathbb{K}_e = \frac{1}{2}CV^2 = \frac{1}{2}C\dot{\psi}^2 \tag{4}$$

$$T_e = \frac{1}{2}L(x)i_L{}^2 - i_s\psi = \frac{1}{2L(x)}\psi^2 - i_s\psi \tag{5}$$

$$\mathbb{K}_m = \frac{1}{2}m\dot{x}^2, \tag{6}$$

$$\mathbb{T}_m = \frac{1}{2}kx^2 - xf(t) \tag{7}$$

$$P_e = -\frac{\dot{\psi}^2}{2R} , P_m = -\frac{b\dot{x}^2}{2} \tag{8}$$

By substituting (4)–(8) into (3):

$$\mathcal{L} = \frac{1}{2}C\dot{\psi}^2 + \frac{1}{2}m\dot{x}^2 - \left(\frac{1}{2L(x)}\psi^2 - i_s\psi + \frac{1}{2}kx^2 - xf(t)\right) \tag{9}$$

For the mechanical subsystem, the Lagrange equation can be written as:

$$\frac{d}{dt}(m\dot{x}) + \frac{\partial\left(\frac{1}{2}kx^2 - xf(t) + \frac{1}{2L(x)}\psi^2\right)}{\partial x} = -b\dot{x} \rightarrow m\ddot{x} + kx - f(t) + \frac{\psi^2}{2}\frac{\partial\left(L(x)^{-1}\right)}{\partial x} = -b\dot{x} \rightarrow$$
$$m\ddot{x} + kx + b\dot{x} - \frac{\psi^2}{2L^2(x)}\frac{\partial(L(x))}{\partial x} = f(t) \rightarrow m\ddot{x} + kx + b\dot{x} = f(t) + \frac{\psi^2}{2L^2(x)}\frac{\partial(L(x))}{\partial x} \tag{10}$$

From the right-hand side of the above, the electromagnetic force applied to the mass by the electromagnet is obtained as follows.

$$F_{em} = \frac{\psi^2}{2L^2(x)}\frac{\partial(L(x))}{\partial x} \tag{11}$$

The system has two degrees of freedom, thus for the electromagnetic part:

$$\frac{d}{dt}\left(C\dot{\psi}\right) + \frac{\partial\left(\frac{\psi^2}{2L(x)} - i_s\psi\right)}{\partial\psi} = -\frac{\dot{\psi}}{R} \rightarrow C\ddot{\psi} + \frac{\dot{\psi}}{R} + \frac{1}{L(x)}\psi = i_s \overset{C=0}{\rightarrow} \frac{\dot{\psi}}{R} + \frac{1}{L(x)}\psi = i_s \tag{12}$$

Note that for $C = 0$: $i_c = C\frac{dV}{dt} = 0$ equivalent to an open circuit.

2.2. Electromagnetic Subsystem

In this work the magnetic circuit solution is adopted, based on Maxwell's equations and the correspondence of the differential equations for linear magnetization problems with those for steady current problems. This is only true in the special case where boundary conditions for the two cases are identical [26]. In order to simplify the analysis of magnetic circuits and make them suitable and accurate enough for engineering applications, assumptions must be made. Initially, the terms of the Maxwell equations containing time-varying electric fields are neglected and the system is regarded as magneto quasi-static. The second assumption refers to the magnetic circuit, which is thought of as a structure of high-permeability materials. This results in the magnetic flux to flow entirely through the path defined by the structure.

An electromagnet featuring a solenoid around a horseshoe-shaped core, an air gap and a moving mass is shown in Figure 2.
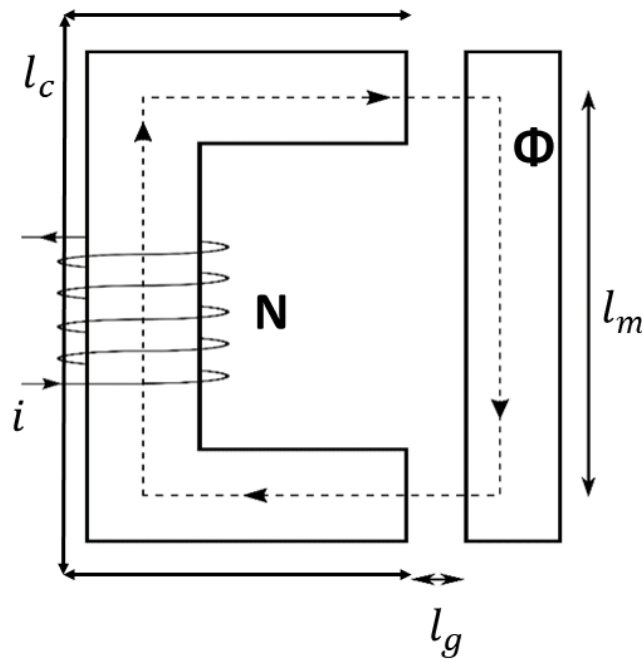
Figure 2. Electromagnet featuring an air gap and a moving mass.

The magnetic reluctances $\Re$ of the circuit are defined as:

$$\Re_c = \frac{l_c}{A_c \mu_0 \mu_c} \tag{13}$$

$$\Re_m = \frac{l_m}{A_r \mu_0 \mu_m} \tag{14}$$

$$\Re_g = \frac{l_g}{A_g \mu_0} \tag{15}$$

where $A_m = A_c = A_g = A$.

The magnetomotive force (mmf) drives a magnetic flux Φ through the magnetic reluctances of the core, air gap and moving mass. From (13), (14) and (15) it is clear that the magnetic reluctance depends on the material permeability. High magnetic permeability can result in small reluctances, in fact much smaller than that of the air gap. The resulting magnetic field stores energy in the air gap equal to:

$$E = \frac{1}{2\mu_0} B^2 A_g l_g \tag{16}$$

The force in the gap acting on the mass is given by the rate of change of energy with gap length, thus:

$$F_{em} = \frac{dE}{dl_g} \overset{(11)}{\rightarrow} F_{em} = \frac{1}{2\mu_0} A_g B^2 \tag{17}$$

It is known that:

$$B = \frac{Ni_L}{A(\Re_c + \Re_m + 2\Re_g)} \tag{18}$$

thus we can write (17) as:

$$F_{em} = \frac{1}{2\mu_0} \frac{N^2 {i_L}^2}{A_g (\Re_c + \Re_m + 2\Re_g)^2} \tag{19}$$

or alternatively:

$$F_{em} = \frac{N^2 A \mu_0 {\mu_c}^2 {\mu_m}^2}{\left(l_c \mu_m + \mu_c l_m + 2\mu_c \mu_m x(t)\right)^2} i^2 \tag{20}$$

The force, acceleration, velocity and displacement are vectors. The electromagnetic force is acting on the mass minimizing the gap length.

Let us now define the two constants (assuming $\mu_c = \mu_m = \mu$):

$$K_a = \frac{1}{2} N^2 A \mu_0 \tag{21}$$

$$K_b = \frac{l_c \mu_m + l_m \mu_c}{\mu_c \mu_m} \tag{22}$$

We can now write (20):

$$F_{em} = \frac{2 K_a {i_L}^2}{\left(K_b + 2x\right)^2} \tag{23}$$

and (11) as:

$$\ddot{x} = \frac{2 K_a {i_L}^2}{m \left(K_b + 2x\right)^2} - \frac{Kx}{m} - \frac{b\dot{x}}{m} \tag{24}$$

The dynamic equations for the electrical subsystem must be derived as well. Knowledge of the induced voltage and variable inductance are key for both sensory and actuating applications.

$$V_L = \frac{d(L(x) i_L)}{dt} \rightarrow V_L = L(x) \frac{di_L}{dt} + i_L \frac{dL(x)}{dx} \frac{dx}{dt} \tag{25}$$

It is known that:

$$\psi = \frac{N i_L(t)}{\left(\frac{l_c}{A_c \mu_0 \mu_c} + \frac{l_m}{A_r \mu_0 \mu_m} + 2\frac{x(t)}{A_g \mu_0}\right)} \tag{26}$$

and:

$$L = \frac{N\psi}{i_L} \rightarrow$$

$$L = \frac{N^2}{\left(\frac{l_c}{A_c \mu_0 \mu_c} + \frac{l_m}{A_r \mu_0 \mu_m} + 2\frac{x(t)}{A_g \mu_0}\right)} = \frac{N^2 A_c \mu_0 \mu_c \mu_m}{l_c \mu_m + \mu_c l_m + 2\mu_c \mu_m x(t)} \tag{27}$$

We assume:

$$\xi_0 = N^2 A_c \mu_0 \mu_c \mu_m, \quad \xi_1 = l_c \mu_m + \mu_c l_m, \quad \xi_2 = 2\mu_c \mu_m \tag{28}$$

and (27) can now be written as:

$$L(x) = \frac{\xi_0}{\xi_1 + \xi_2 x(t)} \tag{29}$$

2.3. Electromechanical System Simulation

The dynamical electromechanical system described in this work is simulated using Matlab [27], in particular:

- Matlab Simulink Simscape;
- Matlab Simulink;

2.3.1. Simscape Implementation

Matlab Simscape uses physical blocks to model complex systems (Figure 3). Each block models an item which is interconnected with others, forming a diagram equivalent to the mathematical model of the system. The interaction of the blocks is accomplished through signal ports permitting non-directional energy exchange. Additional signal ports

are available to the blocks in order to specify initial conditions and enhance the model's computational speed. Blocks are available for both active and passive elements depending on their ability to deliver energy to the system, store it, or dissipate it. Results of the simulated system are shown in Figure 4.
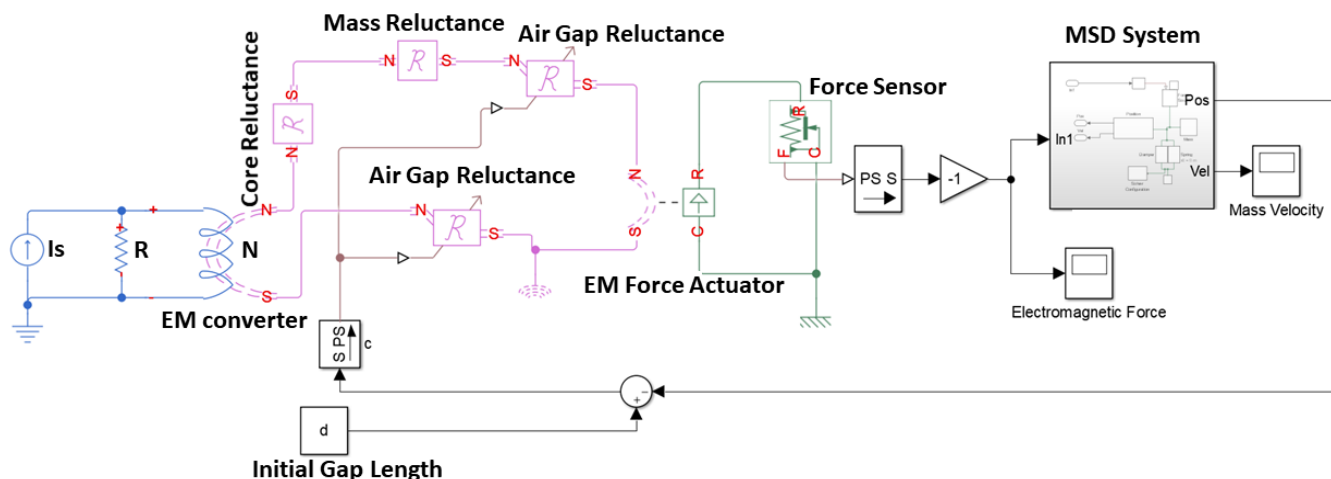

Figure 3. Simulink Simscape implementation of the electromechanical oscillator using the analytical expressions.
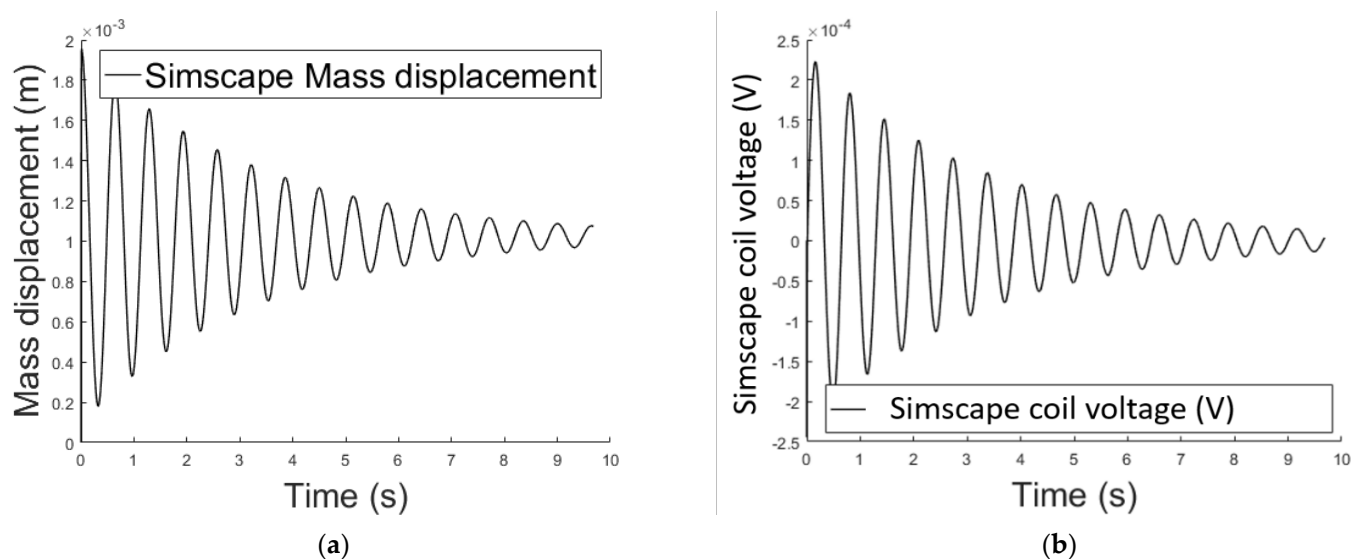

Figure 4. (**a**) Air gap length variation over time using the Simscape implementation; (**b**) variable voltage across the electromagnet's coil over time using the Simscape implementation.

2.3.2. Simulink Implementation

Matlab Simulink allows dynamical modelling using a graphic environment throughout signal blocks. These blocks are complete only when input and output signals are fully defined. The magnetic force function is built and integrated into the Simulink model using the approximation described in (29) for the inductance.

Figures 5 and 6 shows the air gap length and the Simulink coil voltage over a simulation time of 10 s, for the implementation in Figure 5. Initially the output voltage is equal to zero, since the system is in equilibrium (no spring deformation). When a current i is applied to the circuit an attractive force is pulling the mass toward the electromagnet (Figure 2), changing the air gap length thus inducing voltage on the inductance. When the spring force overcomes the magnetic force the mass moves away from the electromagnet.

Motion on the positive x direction induces a positive voltage (negative x direction motion results in negative induced voltage).
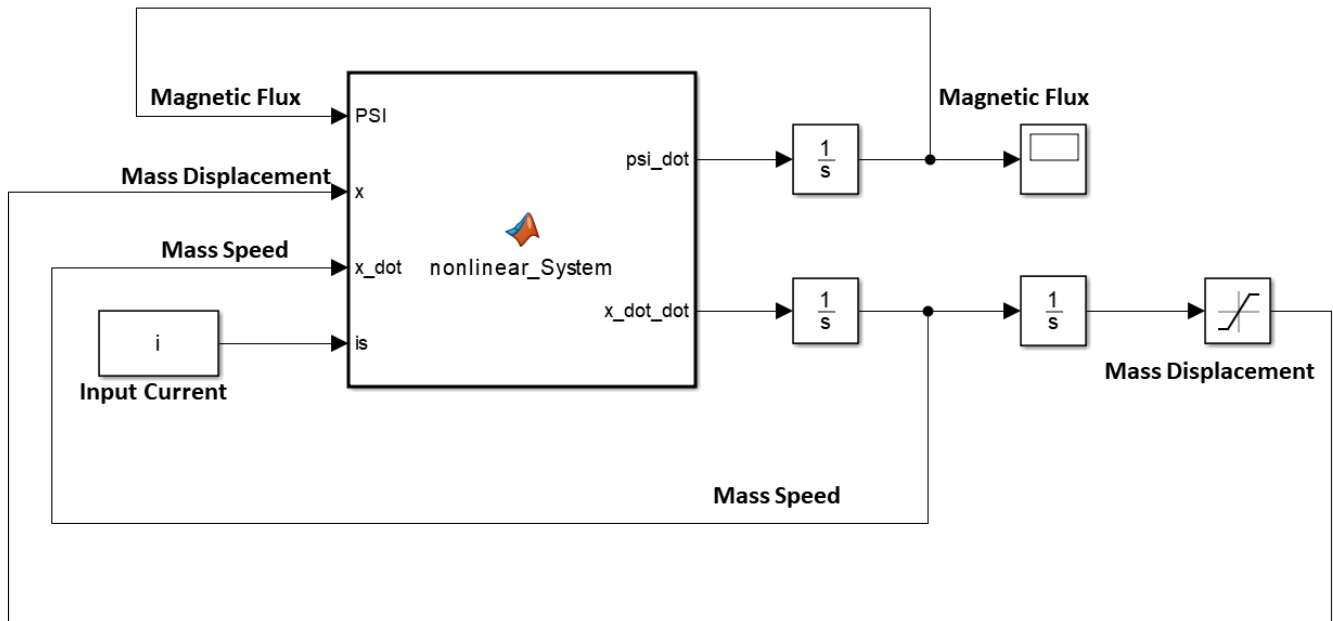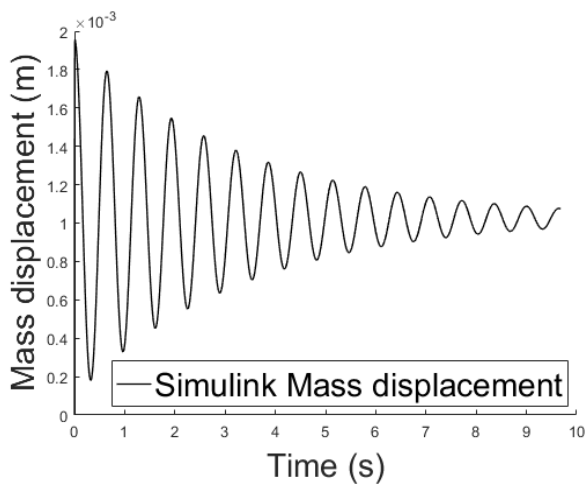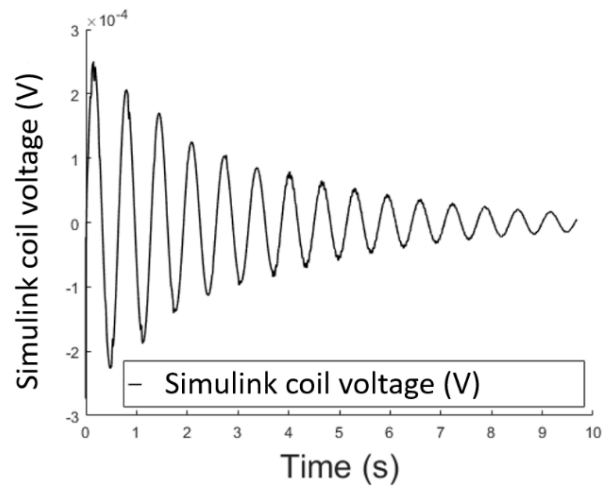

Figure 5. Simulink implementation of the electromechanical oscillator.

Figure 6. (**a**) Air gap length variation over time using the analytical Simulink implementation; (**b**) variable voltage across the electromagnet's coil over time using the analytical Simulink implementation.

2.3.3. Methods Comparison

Next the two methods described above to perform the dynamic simulation of the system are compared. Figure 7 confirms that the results are comparable hence both methods are suitable.
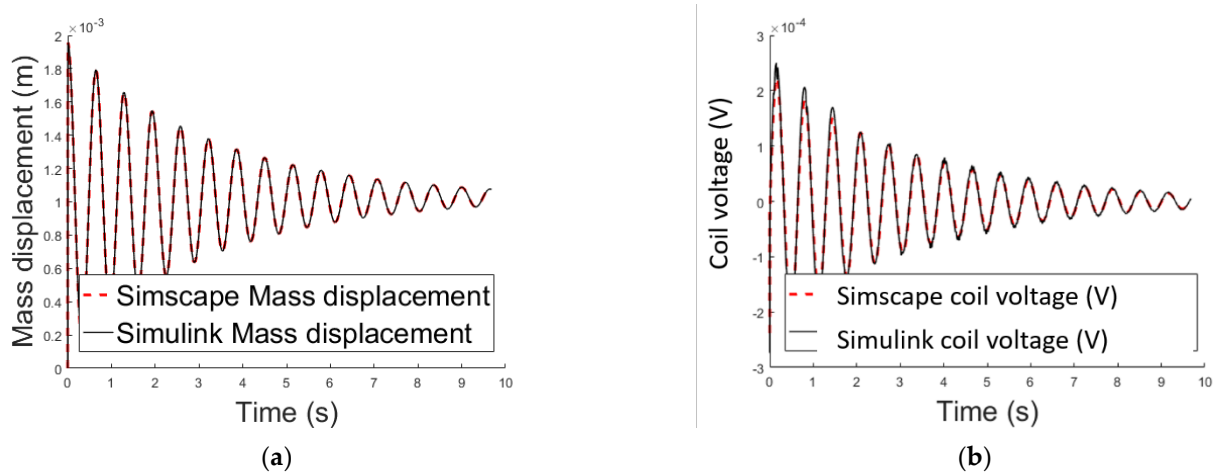
Figure 7. (a) Air gap length variation over time methods comparison; **(b)** variable voltage across the electromagnet's coil over time methods comparison.

3. State-Space Problem Formulation

The state-space model is defined in terms of the derivatives of the states and its representation is given below:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du \tag{30}$$

In order to formulate the state equations of the electro-magneto-mechanical oscillator shown in Figure 1, constraints limiting the system according to its physical and performance characteristics must be defined. For the inductance, we assume the expression described by (29), derived using the magnetic circuit approximation. In favor of simplicity:

$$L = \frac{\xi_0}{\xi_1 + \xi_2 x(t)} \tag{31}$$

where

$$\xi_0 = N^2 A_c \mu_0 \mu_c \mu_m \tag{32}$$

$$\xi_1 = l_c \mu_m + \mu_c l_m \tag{33}$$

$$\xi_2 = 2\mu_c \mu_m \tag{34}$$

Figure 1 shows the origin of the mechanical coordinate (x), assumed equivalent to the neutral position of the spring while that of the electromagnetic system (z) at the position where the mass is marginally in contact with the magnet armature, such that:

$$z = d - x \ [m] \tag{35}$$

thus:

$$-d < x < d \ [m] \tag{36}$$

Based on the expression derived for the electromagnetic force earlier where the magnetic flux and consequently the current appears squared, one can easily derive that the force will be unidirectional, and always attractive, regardless of the electric current's sign:

$$F_{em} \geq 0 \ [N] \tag{37}$$

Additionally, given that we want to avoid any physical contact in the system, the velocity of a moving part must be controlled and constrained as follows [28]:

$$-s(d - x) < \dot{x} < s(d - x) \ [m/s] \tag{38}$$

where s is a constant such that $\dot{x} \in [-5,5] \ [m/s]$.

According to Kirchhoff's law, Newton's second law and the Lagrangian dynamics analysis presented earlier, with the electromagnetic force acting on the moving mass the state equations are:

$$\dot{\psi} = Ri_s - \frac{R}{L(x)}\psi$$
$$\dot{\psi} = Ri_s - \frac{R\xi_0}{\xi_1 + \xi_2 x(t)}\psi = \langle(x, \dot{x}, \psi)\rangle \tag{39}$$

$$\ddot{x} = \frac{\xi_2 \psi^2}{2m\xi_0} - \frac{k}{m}x - \frac{b}{m}\dot{x} = \{(x, \dot{x}, \psi)\} \tag{40}$$

with states: the mass potion x, the mass velocity $\dot{x}$ and the magnetic flux ψ.

Assuming the voltage across the inductance to be the systems' output:

$$V_L = \dot{\psi} = Ri_s - \frac{R\xi_0}{\xi_1 + \xi_2 x(t)}\psi = \langle(x, \dot{x}, \psi)\rangle \tag{41}$$

(39)–(41) represent the dynamics of a nonlinear system thus linearization is applied around some equilibrium point $(x_0, \dot{x}_0, \psi_0, i_{s0})$ using Taylor expansion:

$$\sqcup(x, \dot{x}, \psi) = \sqcup(x_0, \dot{x}_0, \psi_0, i_{s0}) + \sqcup'_x(x_0, \dot{x}_0, \psi_0, i_{s0})(x - x_0) + \sqcup'_{\dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0})(\dot{x} - \dot{x}_0) + \\ \sqcup'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0})(\psi - \psi_0) + \sqcup'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0})(i_s - i_{s0}) , \tag{42}$$

where:

$$\sqcup'_x(x_0, \dot{x}_0, \psi_0, i_{s0}) = \frac{\partial\langle}{\partial x}(x_0, \dot{x}_0, \psi_0, i_{s0}) \ or \ \frac{\partial\{}{\partial x}(x_0, \dot{x}_0, \psi_0, i_{s0}) \tag{43}$$

$$\sqcup'_{\dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0}) = \frac{\partial\langle}{\partial \dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0}) \ or \ \frac{\partial\{}{\partial \dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0}) \tag{44}$$

$$\sqcup'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0}) = \frac{\partial\langle}{\partial \psi}(x_0, \dot{x}_0, \psi_0, i_{s0}) \ or \ \frac{\partial\{}{\partial \psi}(x_0, \dot{x}_0, \psi_0, i_{s0}) \tag{45}$$

and for the equilibrium point $\dot{x}_0 = 0$, $\ddot{x}_0 = 0$ and,

$$\psi_0 = \sqrt{\frac{2\xi_0 K x_0}{\xi_2}} \tag{46}$$

$$i_{s0} = \frac{\psi_0}{L(x_0)} \tag{47}$$

By substituting (39)–(41) into (43)–(45):

$$\langle'_x(x_0, \dot{x}_0, \psi_0, i_{s0}) = -R\psi_0\frac{\xi_2}{\xi_0} \tag{48}$$

$$\langle'_{\dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0}) = 0 \tag{49}$$

$$\langle'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0}) = -\frac{R(\xi_1 + \xi_2 x_0)}{\xi_0} \tag{50}$$

$$\{'_x(x_0, \dot{x}_0, \psi_0, i_{s0}) = -\frac{k}{m} \tag{51}$$

$$\{'_{\dot{x}}(x_0, \dot{x}_0, \psi_0, i_{s0}) = -\frac{b}{m} \tag{52}$$

$$\{'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0}) = \frac{2\xi_2 \ \psi_0}{m\xi_0} \tag{53}$$

and (39)–(41) are written as:

$$\dot{\psi} = \left(-R\psi_0\frac{\xi_2}{\xi_0}\right)(x - x_0) - \frac{R(\xi_1 + \xi_2 x_0)}{\xi_0}(\psi - \psi_0) + R(i_s - i_{s0}) \tag{54}$$

$$\ddot{x} = -\frac{k}{m}(\mathrm{x} - x_0) - \frac{b}{m}(\dot{x} - \dot{x}_0) + \frac{2\xi_2\,\psi_0}{m\xi_0}(\psi - \psi_0) \tag{55}$$

$$V_L = \left(-\mathrm{R}\psi_0\frac{\xi_2}{\xi_0}\right)(\mathrm{x} - x_0) - \frac{R(\xi_1 + \xi_2 x_0)}{\xi_0}(\psi - \psi_0) + R(i_s - i_{s0}) \tag{56}$$

We define as state variables the displacement x, the velocity $\dot{x}$, the magnetic flux ψ and adopt the variable transformation $\delta x = x - x_0\ \delta\dot{x} = \dot{x} - \dot{x}_0$, $\delta\psi = \psi - \psi_0$ and $\delta i = i_s - i_{s0}$. (30) is now written as:

$$\begin{bmatrix} \delta\dot{x} \\ \delta\ddot{x} \\ \delta\dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{k}{m} & -\frac{b}{m} & \frac{2\xi_2\,\psi_0}{mL_0} \\ -\mathrm{R}\psi_0\frac{\xi_2}{\xi_0} & 0 & -\frac{R(\xi_1+\xi_2 x_0)}{\xi_0} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta\dot{x} \\ \delta\psi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ R \end{bmatrix} [\delta i] \tag{57}$$

$$y = \begin{bmatrix} \left(-\mathrm{R}\psi_0\frac{\xi_2}{\xi_0}\right) & 0 & -\frac{R(\xi_1 + \xi_2 x_0)}{\xi_0} \end{bmatrix} \begin{bmatrix} \delta x \\ \delta\dot{x} \\ \delta\psi \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ R \end{bmatrix} [\delta i] \tag{58}$$

and:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ \left(a^2\xi_1 e^{-a(d-x_0)} - \frac{2a^2\xi_1^2 e^{-2a(d-x_0)}}{L(x_0)}\right)\frac{\psi_0^2}{m} - \frac{k}{m} & -\frac{b}{m} & \frac{2a\xi_1 e^{-a(d-x_0)}\,\psi_0}{mL(x_0)^2} \\ \left(\frac{Ra\xi_1 e^{-a(d-x)}}{L(x_0)^2}\psi_0\right) & 0 & -\frac{R}{L(x_0)} \end{bmatrix} \tag{59}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ R \end{bmatrix} \tag{60}$$

$$C = \begin{bmatrix} \left(-\mathrm{R}\psi_0\frac{\xi_2}{\xi_0}\right) & 0 & -\frac{R(\xi_1 + \xi_2 x_0)}{\xi_0} \end{bmatrix} \tag{61}$$

$$D = [R] \tag{62}$$

The partial derivatives described in (43)–(45) to linearize the system are calculated using both the analytical and numerical methods around an equilibrium point point $x_0 = 0.015\ m$ (Table 1).

Table 1. Analytical and numerical values for the linearization partial derivatives.

Partial Derivative	Analytical	Numerical
$\langle'_x(x_0, \dot{x}_0, \psi_0, i_{s0})$	−124.01	−124.86
$\langle'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0})$	−389.97	−390.76
$\langle'_\psi(x_0, \dot{x}_0, \psi_0, i_{s0})$	−910.103	−910.45

and:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ -100 & -0.06 & -910.45 \\ -124.86 & 0 & -390.76 \end{bmatrix} \tag{63}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \tag{64}$$

The system output is the voltage across the inductance in equilibrium position, thus (51):

$$C = [-124.86\ 0 - 390.76] \tag{65}$$

$$D = [2] \tag{66}$$

Table 2 summarizes the state variables, inputs and outputs of the linearized system:

Table 2. State variables, inputs and outputs summary for the linearized system.

States	Input	Output
$\begin{bmatrix} \delta x = x - x_0 \\ \delta \dot{x} = \dot{x} - \dot{x}_0 \\ \delta i_L = i_L - i_{L0} \end{bmatrix}$	$\delta i = i - i_0$	δV_L

Let us now compute the eigenvalues of A to determine whether the linearized open-loop system (without feedback control) is stable. The poles of the transfer function are the solution to:

$$\det(sI - A) = 0 \tag{67}$$

thus: $p_1 = -910.36$, $p_2 = -1.05 + 3.64i$ and $p_3 = -1.05 + 3.64i$. All of three poles lie in the left-half plane making the system stable.

4. System Controllability and Observability

A linear time invariant system $\dot{x} = Ax + Bu$, $x(0) = x_0$ is controllable if over the interval $\left[0, t_f\right]$ a control input $u(t) \ \forall \ t \in \left[0, t_f\right]$ steers the state from x_{init} to x_{t_f} according to:

$$x\left(t_f\right) = e^{At_f} x_{init} + \int_0^{t_f} e^{A(t-\tau)} Bu(\tau) d\tau \tag{68}$$

or in other words an LTI system is controllable at time $t_f > 0$ if for any initial state and for any target state x_{t_f}, a control input $u(t)$ exists that can steer the system from $x(0)$ to x_{t_f} over the defined interval.

It can be shown that an LTI system is controllable if and only if its controllability matrix M has full rank, where $M = [B, \ AB]$. For our system, from (63) and (64):

$$M = \begin{bmatrix} 0 & 0 & 1.2486 \times 10^3 \\ 0 & 1.2486 \times 10^3 & -1.1393 \times 10^6 \\ 2 & -1.8209 \times 10^3 & 1.6578 \times 10^6 \end{bmatrix} \tag{69}$$

The controllability matric M has full rank since rank (M) is equal to number of state variables.

An LTI system is observable at time t_f if the initial state $x(0)$ can be uniquely determined from any given $u(0), \ldots, u\left(t_f - 1\right)$, $y(0), \ldots, y\left(t_f - 1\right)$, where y is the output of the system. By analogy, an LTI system is observable if and only if its observability matrix O has full rank, where $O = [C, \ AC]$. For our system, from (63) and (65):

$$O = \begin{bmatrix} 0 & 0 & 1 \\ 124.8627 & 0 & -910.4529 \\ -1.1368 \times 10^5 & 124.8627 & 8.2892 \times 10^5 \end{bmatrix} \tag{70}$$

The observability matrix O has full rank since rank (O) is equal to number of state variables.

Let's now consider the linearized system as described in Section 3 with state variables the spring displacement x and the velocity $\dot{x}$, and output the inductance voltage V_L. As stated earlier, the system contains an unstable pole with a corresponding damping ratio of -1. A negative damping ratio, called driving force, increases the system oscillation rather than driving the system to stability (damping force).

4.1. Linear Controller Design Using Pole Placement

To stabilize the system around a specific position, a feedback controller is designed using poles placement (Figure 8). In order to implement full-state feedback all state variables must be known to the controller at all times.
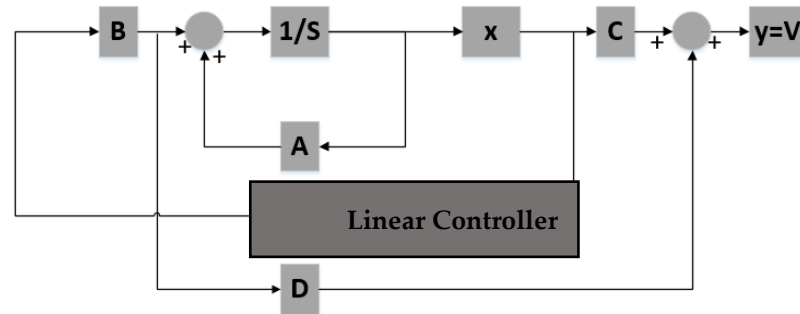

Figure 8. Full-state feedback using poles placement block diagram.

The controller equation is:

$$u = -K_{gain}x \tag{71}$$

where u is the system input, K_{gain} the negative state-feedback gain matrix and x the state variables, and:

$$\dot{x} = Ax + B(-K_{gain}x) = (A - BK_{gain})x \tag{72}$$

$$y = Cx + Du \tag{73}$$

where:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ -100 & -2 & 624.3134 \\ 124.86 & 0 & -910.4529 \end{bmatrix} \tag{74}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \tag{75}$$

$$C = [-124.86 \ 0 - 390.76] \tag{76}$$

$$D = 2 \tag{77}$$

The system specific closed loop A_{cl} matrix is now equal to $(A - BK_{gain})$ and the gain K_{gain} matrix is a 1×2 matrix hence,

$$A_{cl} = \begin{bmatrix} 0 & 1 & 0 \\ -100 & -2 & 624.3134 \\ 124.86 & 0 & -910.4529 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} \begin{bmatrix} K_{gain1} & K_{gain2} & K_{gain3} \end{bmatrix} \tag{78}$$

The eigenvalues of A_{cl} are given by $\det(A_{cl} - \lambda I)$:

$$\det(A_{cl} - \lambda I) = 0 \tag{79}$$

Let us now assume that we would like to place two poles p_1 and p_2 such that the characteristic equation is:

$$(\lambda + p_1)(\lambda + p_2)(\lambda + p_3) = 0 \tag{80}$$

The gain matrix K_{gain} can now be calculated by setting the coefficients equal to each other.

The location of the poles p_1 and p_2 characterize the stability and time-domain performance of the system.

The controller design and the selection of the gain matrix K_{gain} are based on two criteria:

1. overshot < 5%
2. settling time < 0.5 *sec*

For an overshoot less than 5% we have:

$$\exp\left(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right) \quad < 0.05 \rightarrow -\frac{\zeta\pi}{\sqrt{1-\zeta^2}} < \ln(0.05) = -3.00 \rightarrow \frac{\zeta}{\sqrt{1-\zeta^2}} > 0.95 \tag{81}$$
$$\rightarrow \frac{\zeta^2}{1-\zeta^2} > 0.91 \rightarrow \zeta^2 > \frac{0.91}{1+0.91} \rightarrow \zeta^2 > 0.48 \rightarrow \zeta^2 > 0.69$$

and for a settling time (3% error) of less than 0.5 s:

$$t_s = \frac{\ln(0.03)}{\zeta\omega} < 0.5 \rightarrow \zeta\omega > 7 \tag{82}$$

For a second order system with $\zeta < 1$ the poles are given by:

$$-\zeta\omega \mp i\omega\sqrt{1-\zeta^2} \tag{83}$$

The closed loop system showed in Figure 8 is simulated using the following three alternative pole sets:

- $p_{11} = -10 + i10$, $p_{12} = -10 - i10$ and $p_{13} = -50$
- $p_{21} = -20 + i20$, $p_{22} = -20 - i20$ and $p_{23} = -100$
- $p_{31} = -30 + i30$, $p_{32} = -30 - i30$ and $p_{33} = -150$

with gain matrixes:

- $K_{gain1} = [K_{gain11} \ K_{gain12} \ K_{gain13}] = [64.99 \ 0.772 - 421.22]$
- $K_{gain2} = [K_{gain21} \ K_{gain22} \ K_{gain23}] = [115.44 \ 3.54 - 386.22]$
- $K_{gain3} = [K_{gain31} \ K_{gain32} \ K_{gain33}] = [262.01 \ 8.23 - 351.22]$

respectively.

The controller is based on the linearized system while the plant is the non-linear system described by (39)–(41). The current applied to the plant features a small delta with respect to the equilibrium current, $\delta i_{s0} = 0.01 \ A$, to remain in the region where the linearization is valid. Figures 9–11 show the controlled mass displacement for different equilibrium points using poles set as p_{31}, p_{32} and p_{33}.
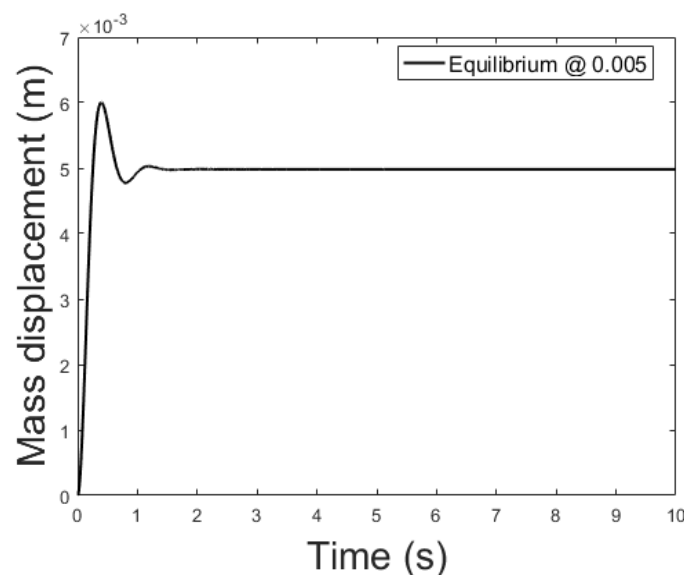

Figure 9. Controlled system output with initial conditions: $x_0 = 0$ m, $\dot{x}_0 = 0$ m/s, $\psi = 0$ Wb and equilibrium at $x = 0.005$.
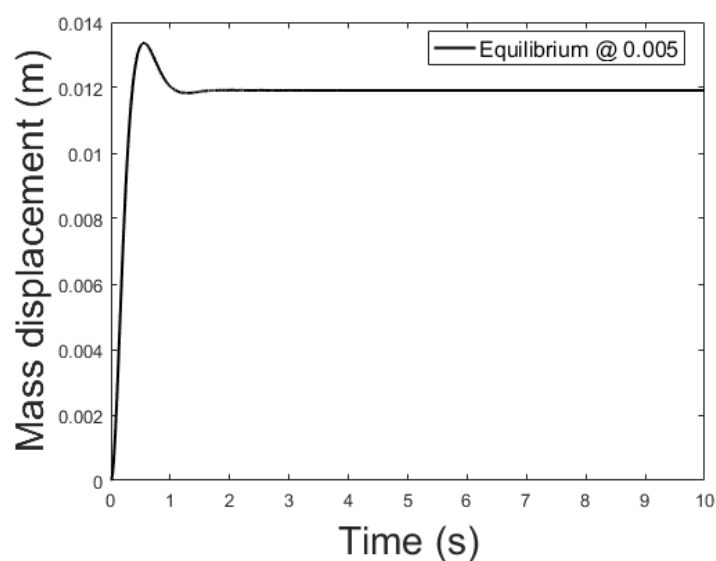
Figure 10. Controlled system output with initial conditions: $x_0 = 0$ m , $\dot{x}_0 = 0$ m/s, $\psi = 0$ Wb and equilibrium at $x = 0.01$.
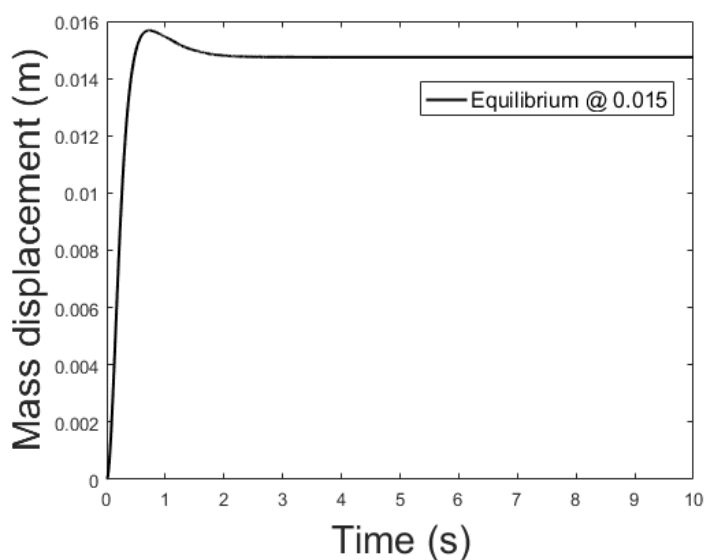

Figure 11. Controlled system output with initial conditions: $x_0 = 0$ m, $\dot{x}_0 = 0$ m/s, $\psi = 0$ Wb and equilibrium at $x = 0.015$.

4.2. Observer Design

The pole placement method described earlier requires knowledge of the system states at all times. In practice, this is very demanding due to unavailable or expensive sensors often driven by high noise as well. Moreover, some types of application require the system to operate as a sensor rather than as an actuator. Given the proven observability of the studied system, a state observer is designed to overcome the issue. A state observer is capable of estimating and observing the system states regardless of whether some state variables are available for direct measurement or not. The only measurable output for the hereby-described electromechanical oscillator is the inductance voltage. The observer mimics the model of the plant, except for an additional term, introduced to compensate for inaccuracies in the matrices A and B. This new term, called estimation error, is equal to the difference between the estimated output value and the measured value.

The observer equations are:

$$\dot{\hat{x}} = A\hat{x} + Bu + \mathfrak{L}(y - \hat{y}) = A\hat{x} + Bu + \mathfrak{L}(y - \hat{C}x - Du)$$
$$= (A - \mathfrak{L}C)\hat{x} + (B - \mathfrak{L}D)u + \mathfrak{L}y \tag{84}$$

$$\hat{y} = C\hat{x} - Du \tag{85}$$

where $\hat{x}$ the estimated states, $\hat{y}$ the estimated output and $\mathfrak{L}$ the observer gain matrix.

The system's output is compared to the estimated output. The resulting error is corrected to help the observer converge to the system state values. The observer's error dynamic is given by:

$$\dot{e} = \dot{x} - \dot{\hat{x}} = (A - \mathfrak{L}C)e \tag{86}$$

The input to the open-loop plant is equal to:

$$u = -K_{gain}\hat{x} \tag{87}$$

Thus:

$$\dot{x} = Ax + Bu = Ax + B\left(-K_{gain}x\right) = \left(A - BK_{gain}\right)x - BK_{gain}(\hat{x} - x) \Rightarrow \dot{x}$$
$$= \left(A - BK_{gain}\right)x + BK_{gain}e \tag{88}$$

and:

$$\begin{bmatrix} \dot{x} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - BK_{gain} & BK_{gain} \\ 0 & (A - \mathfrak{L}C) \end{bmatrix} \begin{bmatrix} x \\ e \end{bmatrix} \tag{89}$$

(88) shows that the state dynamics is decoupled from the observer's error e (separation principle) thus the gain K can be computed independently of the observe gain L. The gain matrix K_{gain} is computed, for $p_{31} = -30 + i30$, $p_{32} = -30 - i30$, $p_{33} = -150$ and is equal to $K_{gain} = [262.01\ 8.23 - 351.22]$. For the observer, we select eigenvalues located at least five times further left on the complex plane than the plant eigenvalues, since we want the dynamics of the observer to be much faster than the system itself. In other words, the observer must converge to the plants' states before they converge to zero. Hence we choose: $p_{obs1} = -150 + 6i$, $p_{obs2} = -150 - 6i$ and $p_{obs3} = -750$. Poles placed further apart than that could result in complex systems with large bandwidth, making the system vulnerable to noise effects. The analytical procedure to derive the observer matrix is identical to the one described for the gain matrix K_{gain}. Alternatively, one can use the place command in Matlab and: $\mathfrak{L} = \left[-9.405 \times 10^3\ 1.914 \times 10^3 - 1.29\ 3.23 \times 10^3\right]$. The selection of matrix $\mathfrak{L}$ is a tradeoff between a fast dynamic response and noise reduction.

Figure 12 shows the non-linear plant-linear controller-linear observer block diagram. Figure 13 shows theMatlab Simulink system implementation. The states are now unknown to the controller, hence the observer is used to generate estimates by comparing the plant output to the observer output. These estimates of the states are fed into the linear controller to stabilize the system around an equilibrium point.

Figures 14–16 depicts the system output, with the observer and the linear controller, for the following initial conditions:

- $x_0 = 0.006$ m, $\hat{x}_0 = 0.002$ m
- $\dot{x}_0 = 0$ m/s, $\dot{\hat{x}}_0 = 0.04$ m/s
- $\psi_0 = 0$ Wb, $\hat{\psi}_0 = 0.07$ Wb

and equilibrium at 0.005, 0.010 and 0.015 m.

The observer is a virtual system, designed to approximate the state variables not available for sensing. A properly designed observer has all initial conditions specified. Despite there isn't a standard rule, for this study are chosen such that the initial error is equal to $e = [0.004, -0.04, -0.07]$. Responses of the state variables, for different equilibrium points are depicted in Figures 17–19: it is clear that the estimate states converge to the actual state variables while tracking reasonably well the equilibrium value.
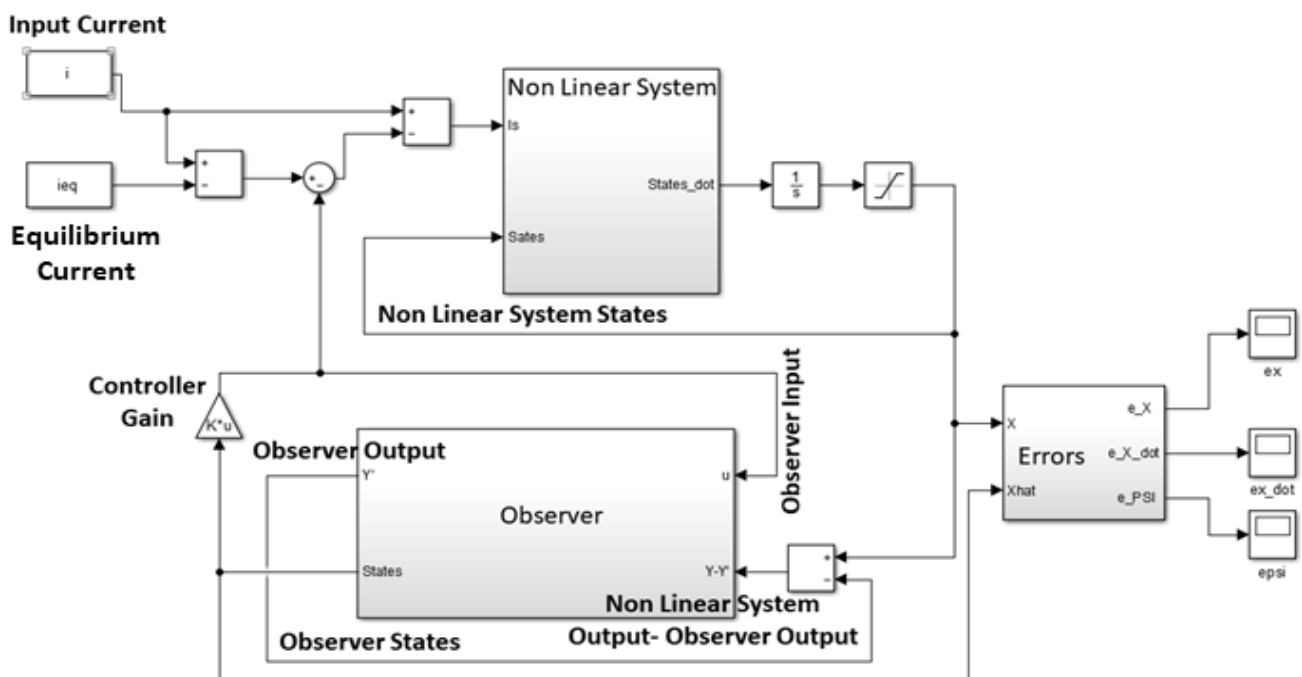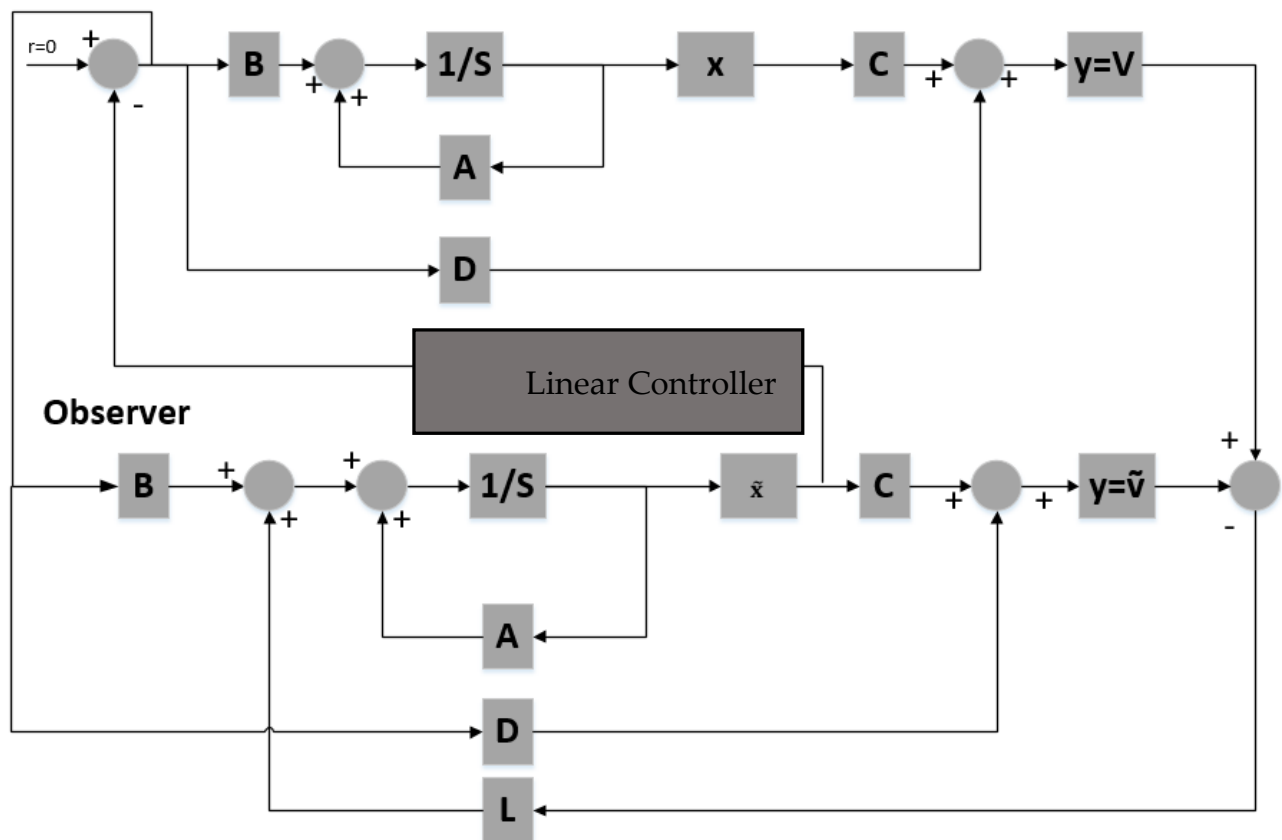
Open Loop plant

Figure 12. Closed loop block diagram.

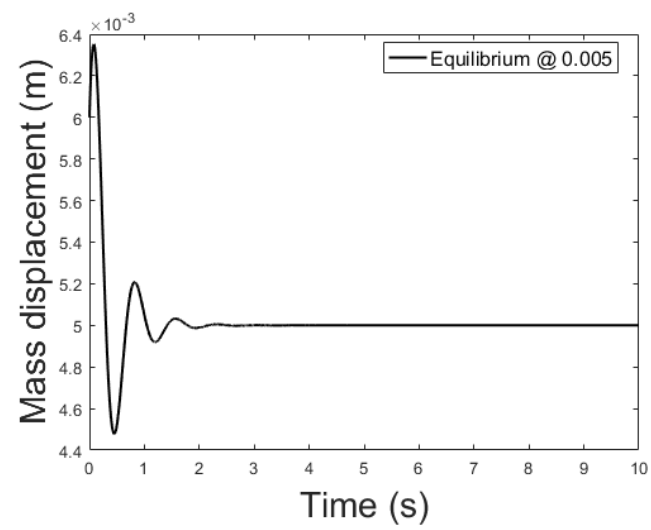**Figure 13.** Simulink implementation of the nonlinear system, controller and observer.

Figure 14. Observer controlled system output with equilibrium at $x = 0.005$.
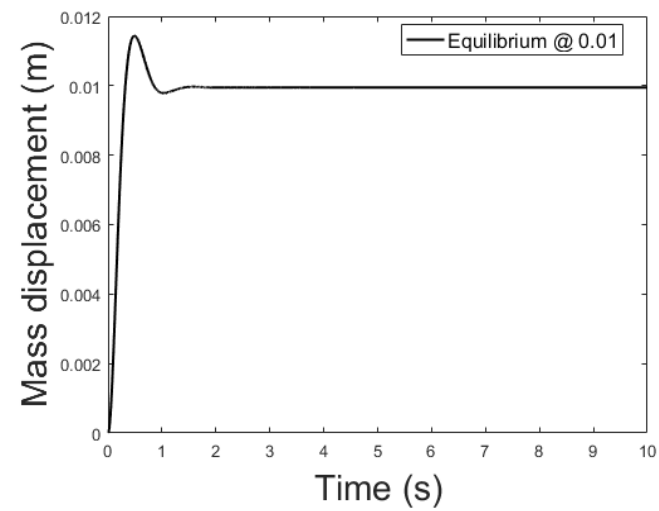

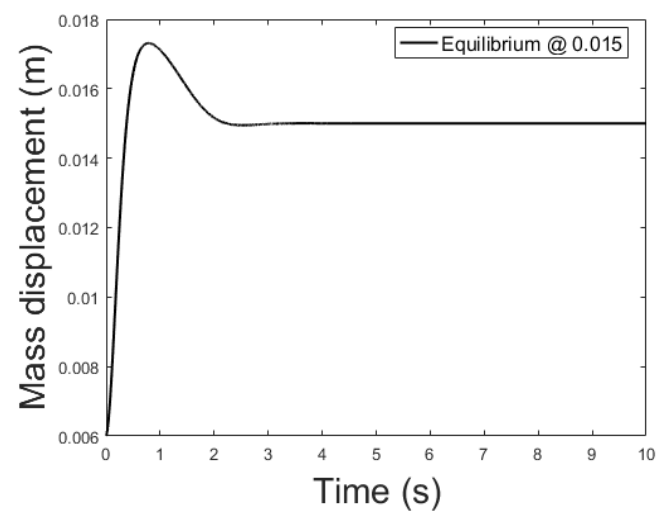**Figure 15.** Observer controlled system output with equilibrium at $x = 0.010$.

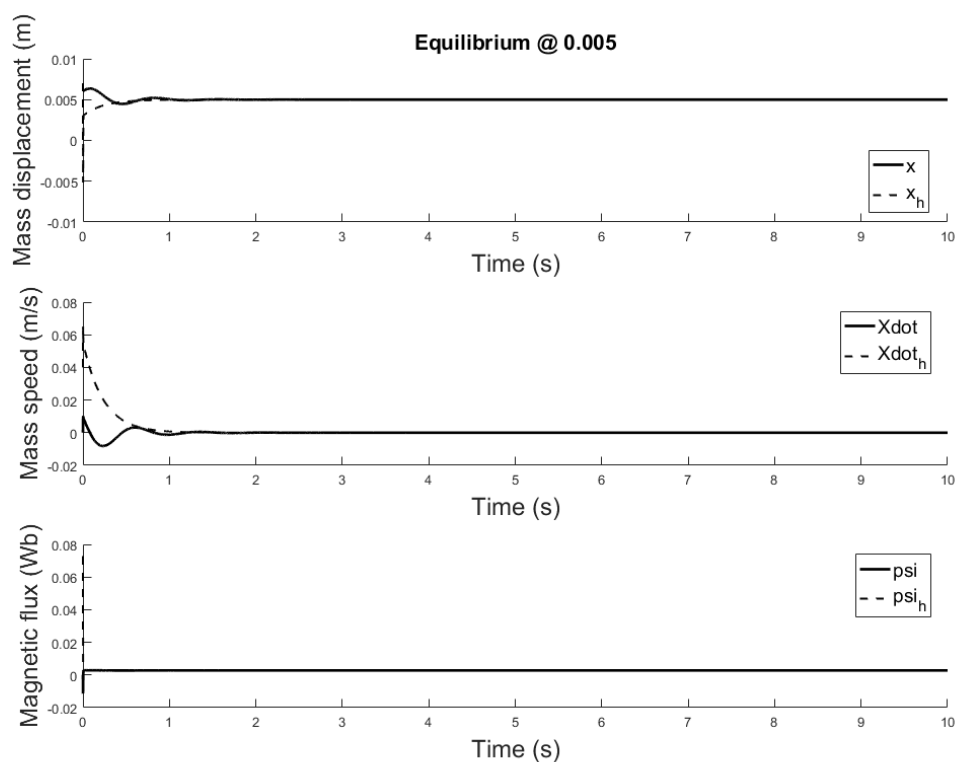**Figure 16.** Observer controlled system output with equilibrium at $x = 0.015$.

Figure 17. Observer and plant state estimates for initial conditions: $x_0 = 0.006$ m, $\hat{x}_0 = 0.001$ m and $\dot{x}_0 = 0$ m/s, $\hat{\dot{x}}_0 = 0.04$ m/s and $\psi = 0$ Wb, $\hat{\psi} = 0.07$ Wb, equilibrium at $x = 0.005$.

Figure 18. Observer and plant state estimates for initial conditions: $x_0 = 0.006$ m, and $\dot{x}_0 = 0$ m/s, $\hat{\dot{x}}_0 = 0.04$ m/s and $\psi = 0$ Wb, $\hat{\psi} = 0.07$ Wb, equilibrium at $x = 0.01$.

Figure 19. Observer and plant state estimates for initial conditions: $x_0 = 0.006$ m, $\hat{x}_0 = 0.001$ m and $\dot{x}_0 = 0$ m/s, $\hat{x}_0 = 0.04$ m/s and $\psi = 0$ Wb, $\hat{\psi} = 0.07$ Wb, equilibrium at $x = 0.015$.
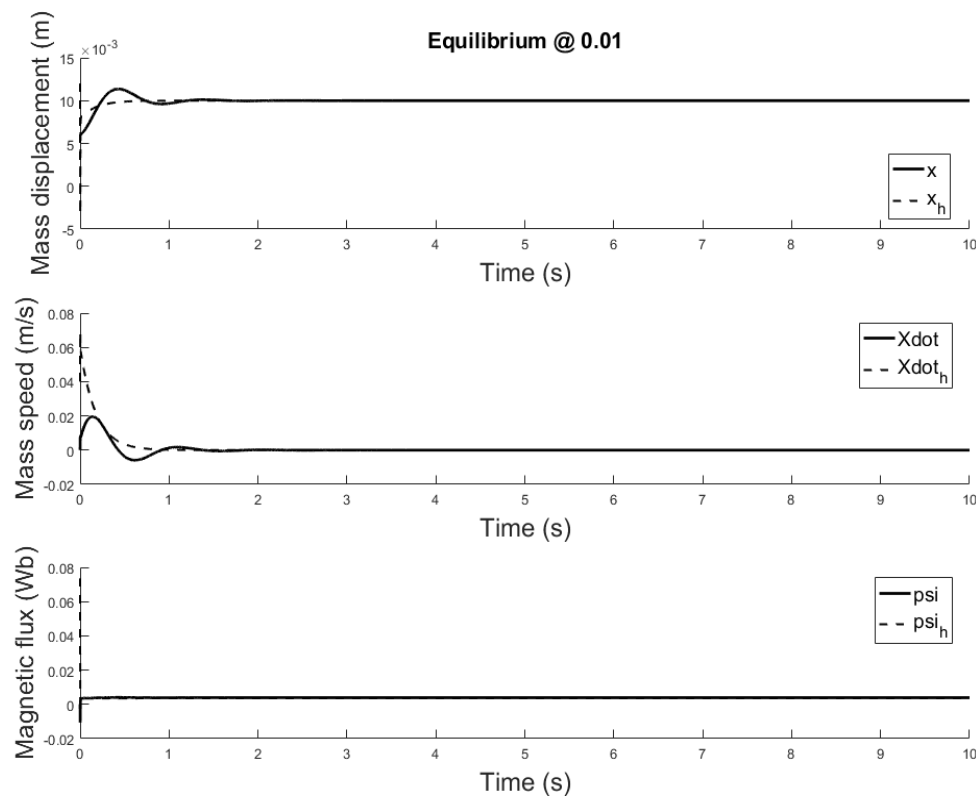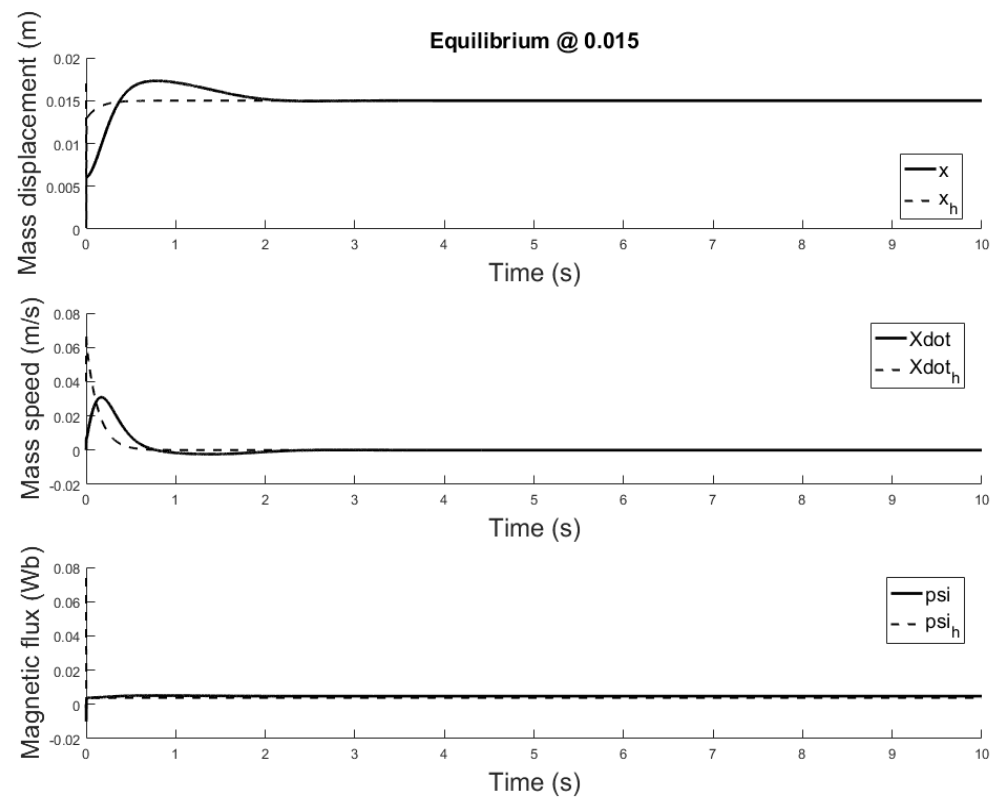
5. Artificial Neural Network (ANN) Representation of the System

An ANN is an adaptive, often non-linear system, that learns to perform a function (an input/output map) when an input and the corresponding desired or target response set is presented to the untrained ANN [29].

An ANN is structured in three layers (Figure 20):

- the input layer, where there is no real processing done, is essentially a "fan-out" layer where the input vector is distributed to the hidden layer;
- the hidden layer, being the computational core of the ANN;
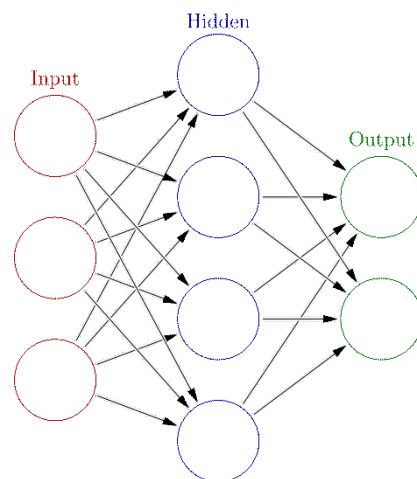- the output layer, which combines all the "votes" of the hidden layer.

Figure 20. Artificial neural network (ANN) architecture diagram.

The general mathematical expression that describes an ANN is [29]:

$$y_{ANN} = (u_{\_n} f_{ANN}(x_{ANN} w_{\_n} + b_{ANN})) + b_{ANN} \tag{90}$$

First, the input x_{ANN} is multiplied by the weight $w_{\_n}$. Second, the weighted input $xw_{\_n}$ is added to the scalar bias b_{ANN} to form the net input n. The bias is similar to a weight, except that it has a constant input of 1. Finally, the net input is passed through the transfer function f_{ANN}, which produces the output y_{ANN}.

The equation can be more detailed and further expanded to [29]:

$$h_{\_j} = f_{ANN}\left(\sum_{i=0}^{p} w_{\_ji} x_{ANN\_i}\right) \tag{91}$$

$$y_{ANN\_k} = \sum_{j=o}^{M} (u_{\_kj} h_{\_j}) \tag{92}$$

with p the number of input nodes, M the number of hidden nodes and K the number of output nodes.

The activation function f_{ANN} is commonly chosen to be one of the following:

- the logistic sigmoid function (Figure 6), commonly abbreviated as logsig,

$$f_{ANN}(z) = \frac{1}{1 + e^{-z}} \tag{93}$$

- the hyperbolic tangent function (Figure 6), commonly abbreviated as tansig,

$$f_{ANN}(z) = \frac{1 - e^{-2z}}{1 + e^{-2z}} \tag{94}$$

The change in error due to output layers weight can be found by Least Mean Square (LSM) algorithms. The output error is given by [29]:

$$E_{\_x} = \frac{1}{2} \sum_{k=1}^{K} (d_{\_k} - y_{\_k})^2 \tag{95}$$

The back-propagating error corresponding to the hidden layer output can be expressed as the partial derivative [29]:

$$\frac{\partial E_{\_x}}{\partial u_{\_kj}} = \frac{\partial E_{\_x}}{\partial y_{\_k}} \frac{\partial y_{\_k}}{\partial u_{\_kj}} \tag{96}$$

Successful training of the network requires calibration of the weights to counteract the error gradient to minimize the output error according to [29]:

$$u_{\_kj} = -\eta \frac{\partial E_{\_x}}{\partial u_{\_kj}} = -\eta \delta y_{\_k} h_{\_j} \tag{97}$$

$$u_{\_kj}^{new} = u_{\_kj}^{old} + \Delta u_{\_kj} \tag{98}$$

$$w_{\_ji} = -\mu \frac{\partial E_{\_x}}{\partial w_{\_ji}} = -\mu \delta h_{\_j} x_{\_i} \tag{99}$$

$$w_{\_ji}^{new} = w_{\_ji}^{old} + \Delta w_{\_ji} \tag{100}$$

5.1. ANN System Configuration

In this section, a feed-forward network organized in layers is proposed to approximate the electro-magneto-mechanical system. The input layer has three neurons, the hidden

layer 10 neurons while the output layer has three neurons. The workflow design has the following steps:

- Collect data
- Create the network
- Configure the network
- Initialize the weights and biases
- Train the network
- Validate the network
- Use the network

5.1.1. Generation of ANN Data

The training data is generated using the system described in Section 4. In order to provide some variance to the system, the input current is modelled as:

$$i = i_0 + I_0 \cos(\omega t) \tag{101}$$

The training data set for the ANN correspond to the source current (input), the inductance voltage (input), the simulation time (input) and the state variables (output).

5.1.2. ANN Implementation and Training

The Matlab Neural Network Toolbox Mathworks (The MathWorks, Inc.) is used to construct the network. The selected training algorithm is the Levenberg–Marquardt with the following samples set aside for:

- training, 70% of the data;
- validation, 15% of the data;
- and the remaining 15% for testing.

5.2. ANN Simulation of the Non-Linear System-Linear Controller-Observer

The system described in Section 4.2 is now been simulated by the trained ANN analogue. Figure 21 shows the ANNs setup and the training tool provided by Matlab Simulink. Figures 22–27 depict the ANN performance for different equilibrium positions: $x = 0.005$ m, $x = 0.01$ m, $x = 0.015$ m.
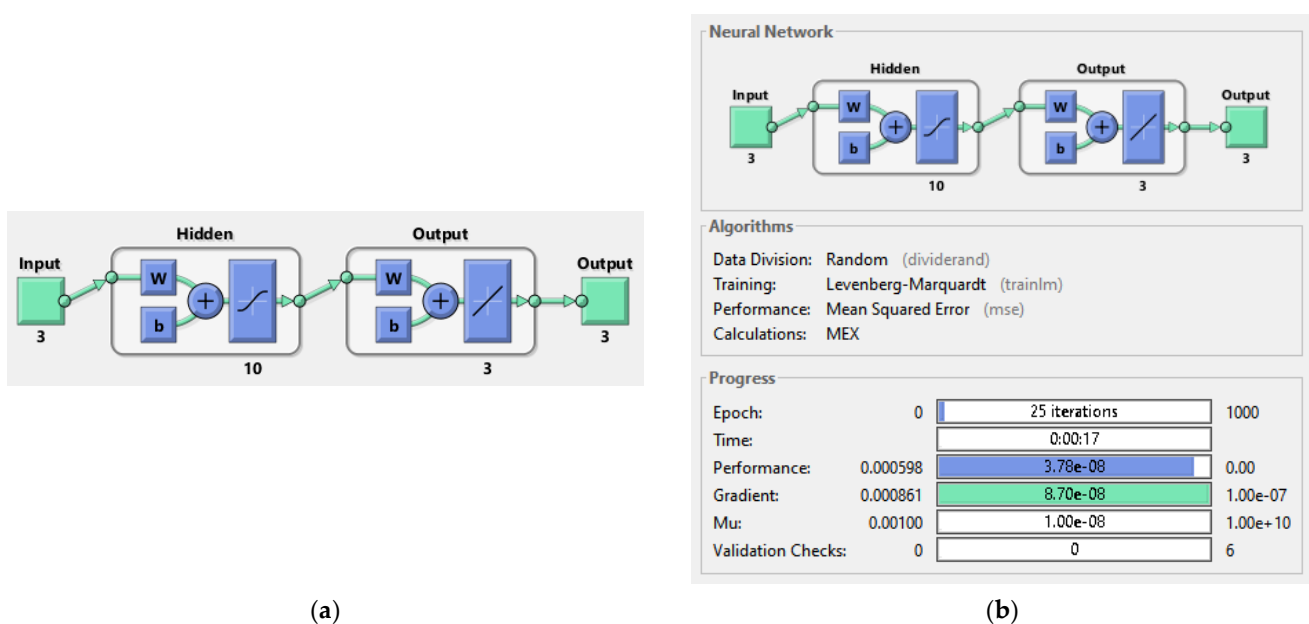

(a)　　　　　　　　　　　　　　　(b)

Figure 21. (**a**): ANN architecture setup; (**b**): Matlab Simulink ANN training tool.
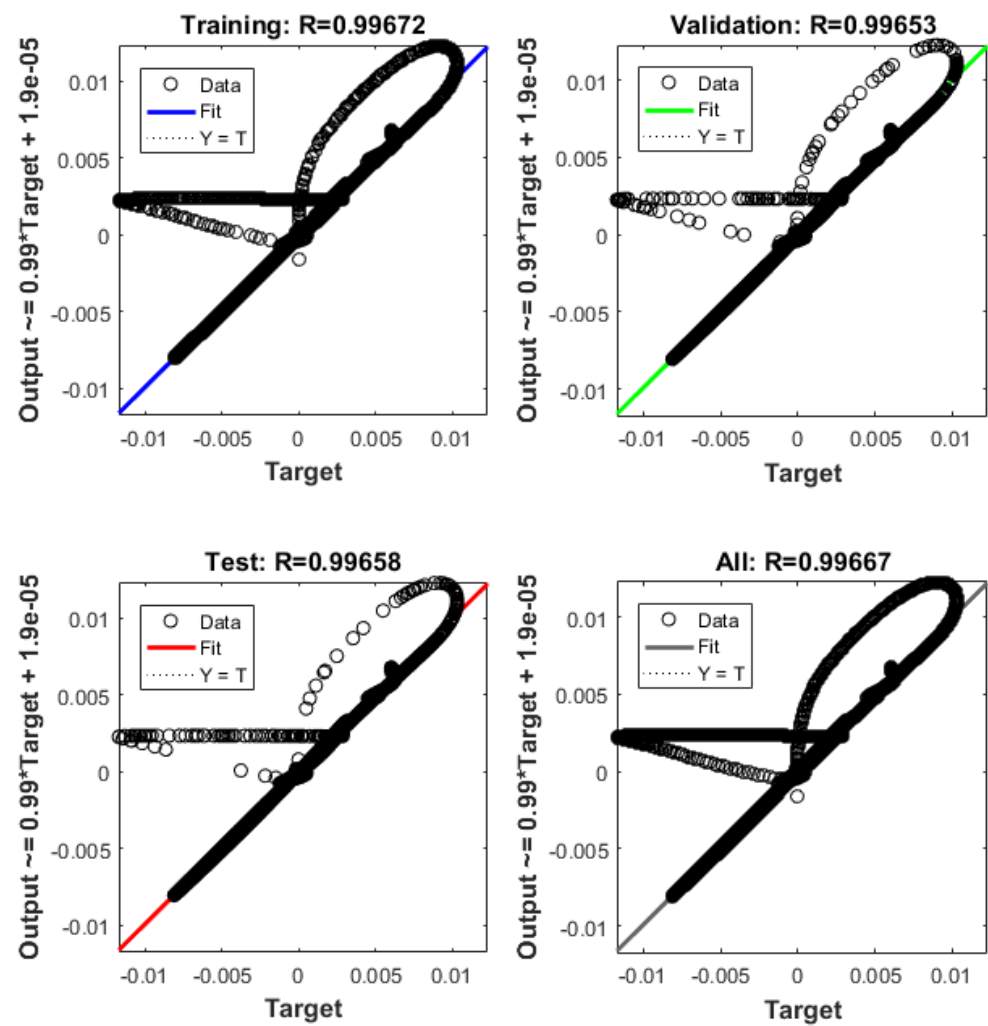
Figure 22. ANN regression plot, equilibrium at $x = 0.005$ m.

Figure 23. (**a**) Exact model mass displacement versus ANN model mass displacement, equilibrium at $x = 0.005$ m; (**b**) exact model mass speed versus ANN model mass speed, equilibrium at $x = 0.005$ m.

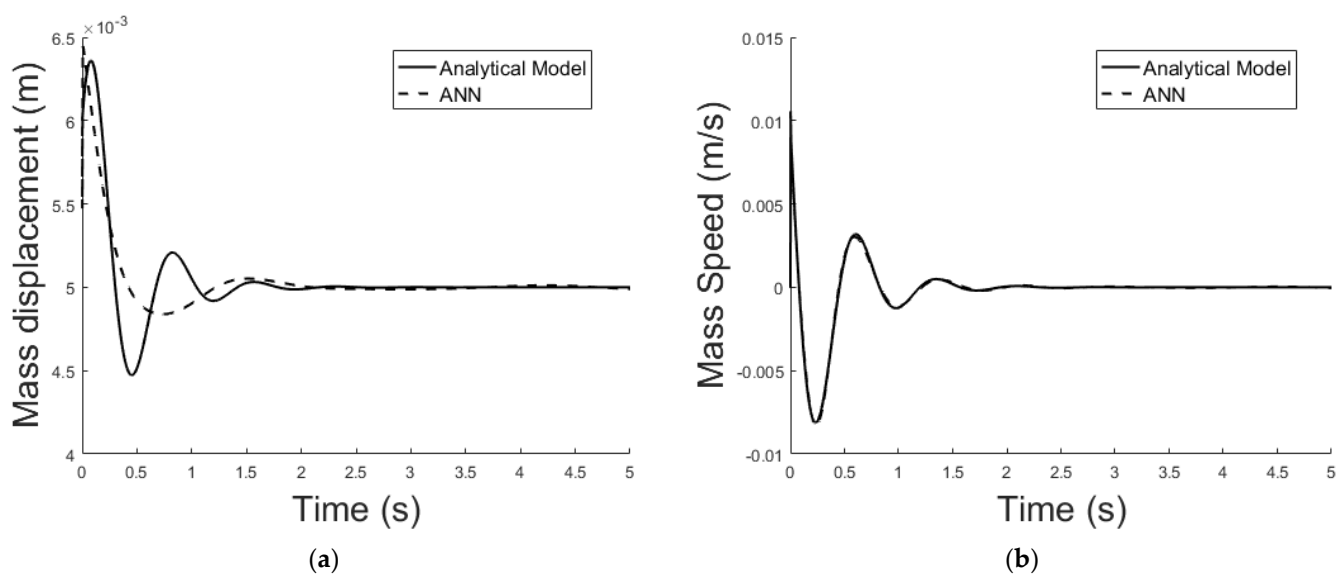**Figure 24.** ANN regression plot, equilibrium at $x = 0.01$ m.

Figure 25. (**a**) Exact model mass displacement versus ANN model mass displacement, equilibrium at $x = 0.01$ m; (**b**) exact model mass speed versus ANN model mass speed, equilibrium at $x = 0.01$ m.

Figure 26. ANN regression plot, equilibrium at $x = 0.015$ m.

Figure 27. (**a**) Exact model mass displacement versus ANN model mass displacement, equilibrium at $x = 0.015$ m; (**b**) exact model mass speed versus ANN model mass speed, equilibrium at $x = 0.015$ m.
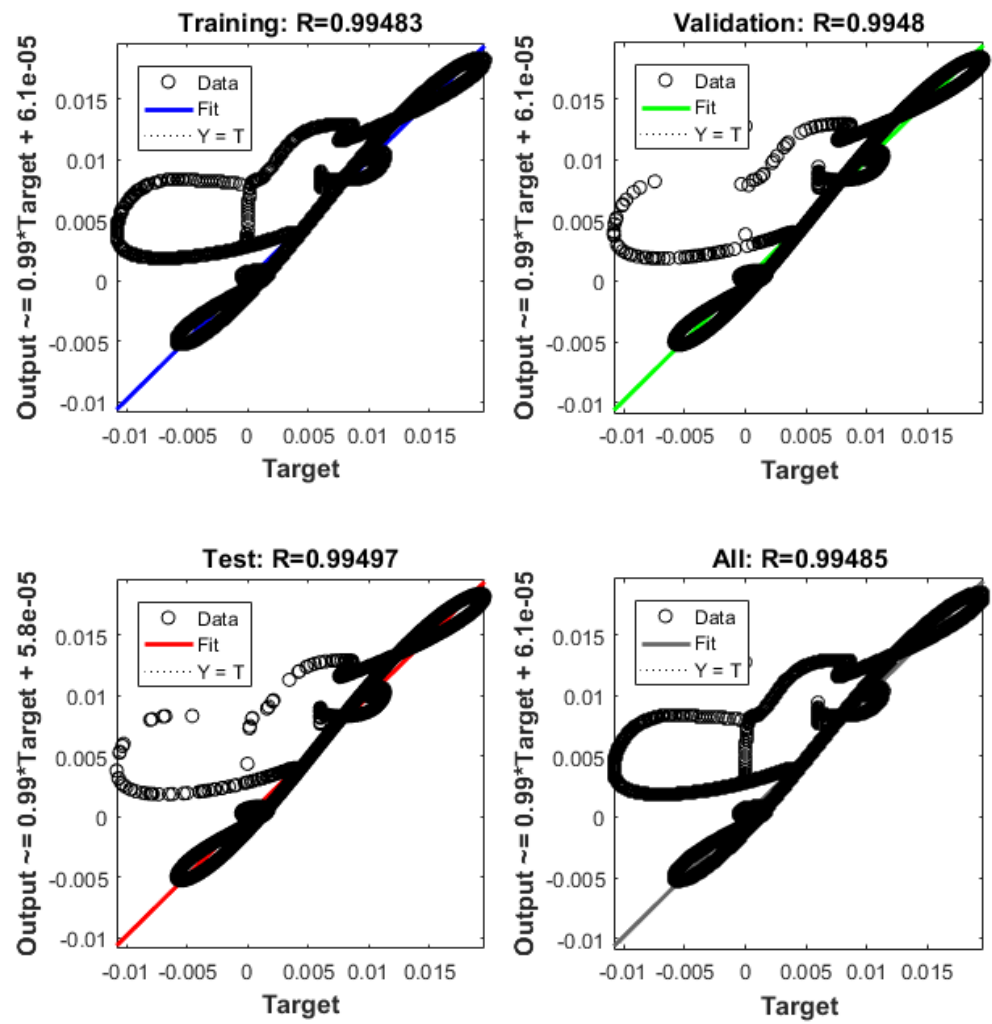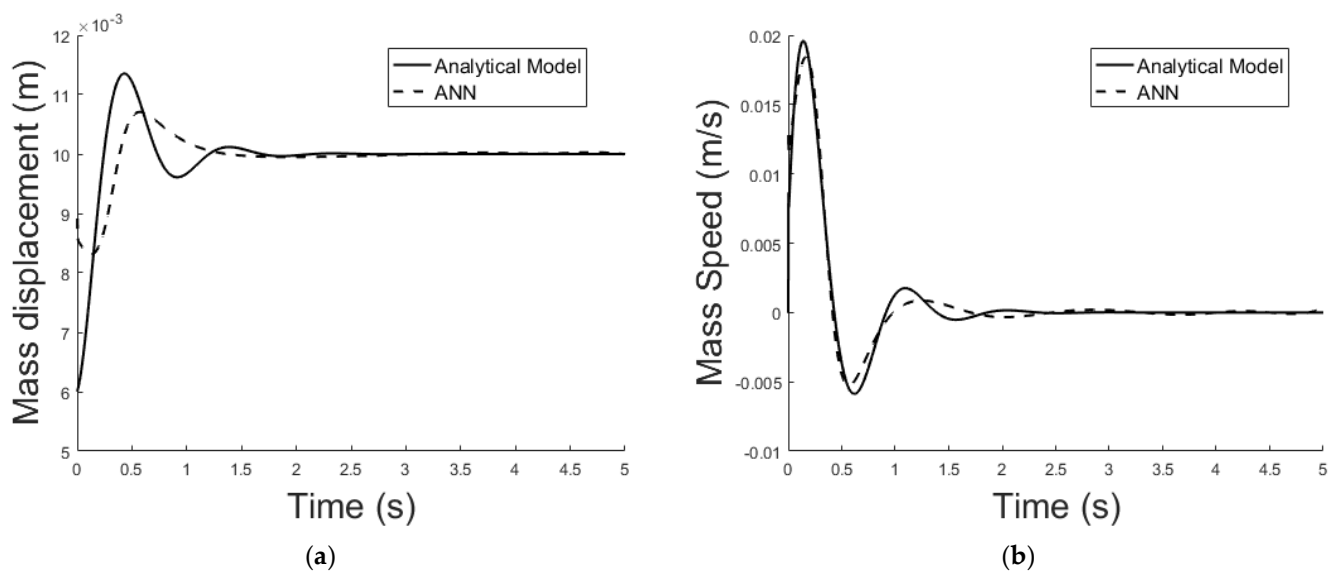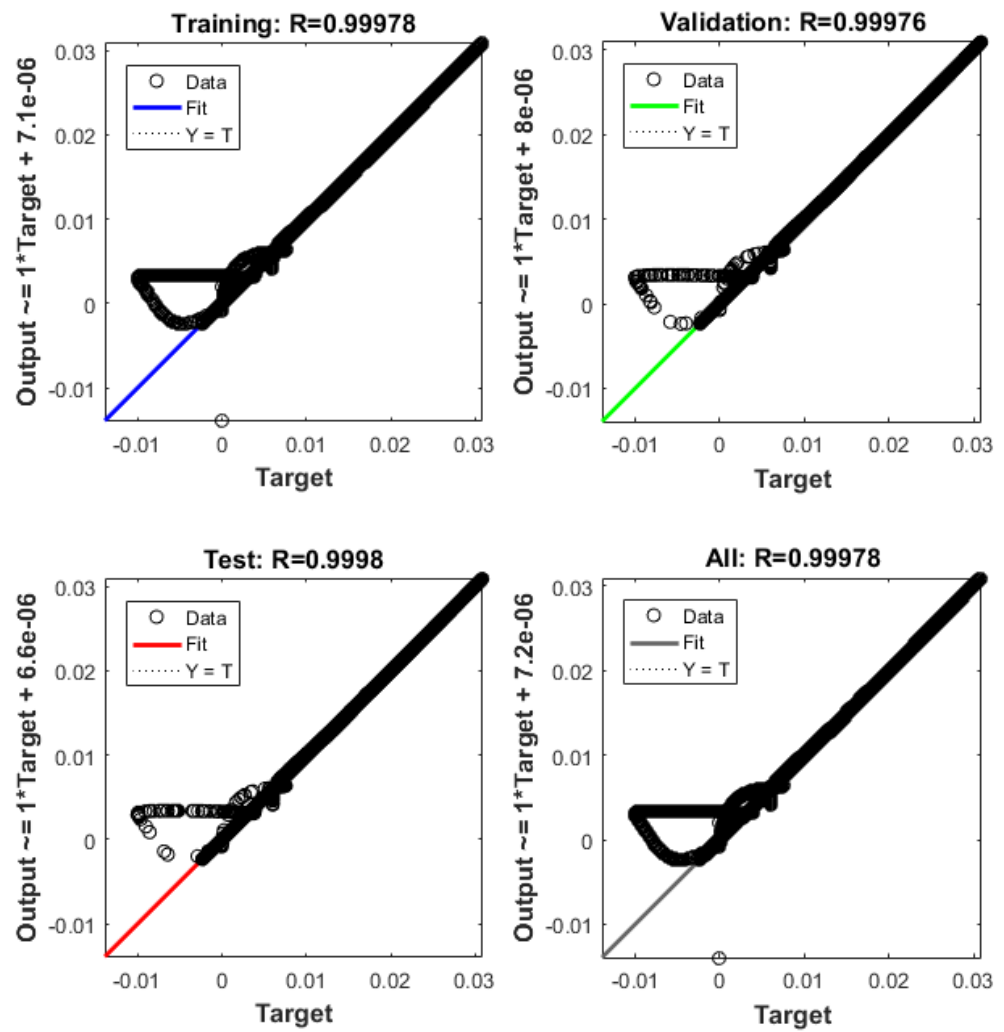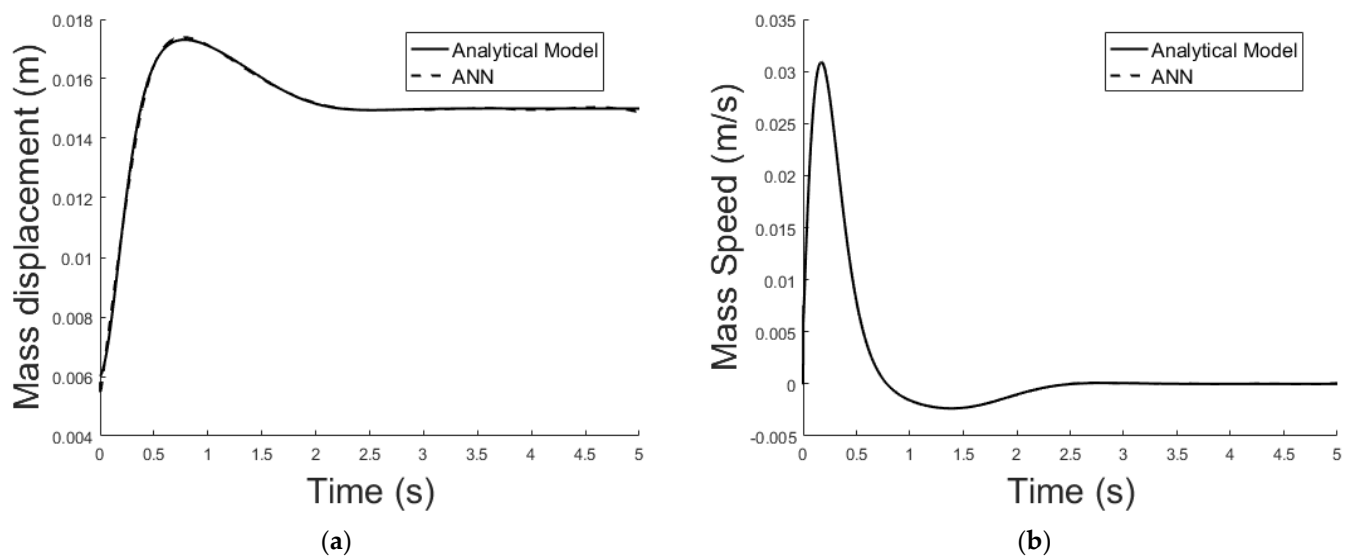
The dashed line (Figures 22, 24 and 26) represents the ideal case where the ANN simulated results are an exact replica of the targets. The solid line represents the best-fit linear regression line between trained net outputs and targets, which in our case is adjacent to the ideal case. All cases indicate a good fit, with the scatter manifesting the data points exhibiting a poor fit. R correlates the targets to the simulated net outputs. If R = 1, there is an exact linear relationship between outputs and targets. If R = 0, no linear relationship between outputs and targets can be inferred. All three cases, for different equilibrium points, show a good approximation of the non-linear plant-linear controller-linear observer system output by the ANN (Figures 23, 25 and 27).

6. Conclusions

The dynamics of an electro-magneto-mechanical system was investigated in this work. The coupled system simulation was developed and comparatively assessed using different modelling techniques in Matalb/Simulink and Matlab/Simscape. Next, the nonlinear state-space problem formulation and the linearization method using partial derivatives were presented. The resulting linearized system proved both controllable and observable, yet marginally stable. To stabilize and control the system, pole-placement with full-state feedback is employed. To implement the full-state feedback a state observer is introduced and designed. Using the models developed and the simulation results, an ANN equivalent has been configured and tuned for the non-linear plant-linear controller-linear observer system. The model is flexible and accurate; specifically, it is able to adapt to different equilibrium conditions (equilibrium position was varied in the range 0.005 m and 0.015 m) and compensates satisfactorily for changes in current source input (sinusoidal reference input). The system may operate as a contactless displacement sensor or actuator, or in combined mode. In ongoing work, the connection of more than one transducer to the same source will be demonstrated using amplitude modulation. The tuned frequency of each transducer will be defined by assigning different values to the capacitor as shown in Figure 1b. Preliminary analysis shows that such transducer arrays can be used in maritime and aerospace applications, where avoiding physical contact is often a sine qua non requirement, to achieve coordinated sensing and actuation of e.g., space structures like in an orbital station to detect and mitigate the vibrational effects of hits by space debris; or in the maritime, for energy harvesting without violating hull integrity with penetrations to implement physical contact. Other applications are possible too, especially if promising theoretical tools like feedback linearization (instead of the local linearization demonstrated here) are employed.

Nomenclature

$\boldsymbol{A}$	System matrix
A	Cross section (m^2)
A_c	Core cross section (m^2)
$\boldsymbol{A}_{cl}$	Closed loop system matrix
A_g	Gap cross section (m^2)
A_m	Mass cross section (m^2)
$\boldsymbol{B}$	Input matrix
B	Magnetic flux density (Wb)
b	Damping constant (Ns/m)
b_{ANN}	ANN scalar bias
$\boldsymbol{C}$	Output matrix
C	Capacitance (F)
$\boldsymbol{D}$	Feedforward matrix
d	Initial gap length (m)
E	Energy (J)
E$_{-x}$	LSM error
e	Observer error
$\dot{e}$	Observer error variation over time
F_{em}	Electromagnetic force (N)
f	Force (N)
$\{$	Function
f_{ANN}	Activation function
G	Conductance (Ω^{-1})
$\langle$	Function
I_0	Current amplitude (A)
i	Current (A)
i_L	Inductance current (A)
i_0	Equilibrium point current (A)
i_s	Source current (A)
i_{s0}	Source current at equilibrium (A)
$\mathbb{K}$	Kinetic energy (J)
K_a	Constant
K$_b$	Constant
$\mathbb{K}_e$	Electrical kinetic energy (J)
K_{gain}	Gain matrix
$\mathbb{K}_m$	Mechanical kinetic energy (J)
k	Spring constant (N/m)
L	Inductance (H)
$\mathcal{L}$	Lagrangian quantity
$\mathfrak{L}$	Observer gain matrix
l_c	Core length (m)
l_g	Gap length (m)
l_m	Mass length (m)
$\boldsymbol{M}$	Controllability matrix
m	Mass (Kg)
N	Number of turns
$\boldsymbol{O}$	Observability matrix
P	Dissipation term (J)
p	Poles
q_i	Independent variables
R	Resistance (Ω)
$\Re$	Magnetic reluctance (H^{-1})
$\Re_c$	Core magnetic reluctance (H^{-1})
$\Re_g$	Gap magnetic reluctance (H^{-1})
$\Re_m$	Mass magnetic reluctance (H^{-1})
s	Constant

$\mathbb{T}$	Potential energy (J)
$\mathbb{T}_e$	Electrical potential energy (J)
$\mathbb{T}_m$	Mechanical potential energy (J)
$\langle$	Taylor expansion function
t_s	Stelling time (s)
$\boldsymbol{u}$	Input control vector
V	Voltage (V)
V_L	Inductor voltage (V)
w_n	ANN weight matrix between hidden and hidden layer
x	Displacement (m)
$\boldsymbol{x}$	State variable vector
$\hat{\boldsymbol{x}}$	Estimated state variable vector
x_{ANN}	ANN input vector
x_0	Equilibrium point displacement (m)
$\dot{x}_0$	Equilibrium point speed (m/s)
$\hat{x}_0$	Observer equilibrium point displacement (m)
$\hat{\dot{x}}_0$	Observer equilibrium point speed (m/s)
$\boldsymbol{y}$	Output vector
$\hat{\boldsymbol{y}}$	Estimated output vector
y_{ANN}	ANN output
z	Neutral position of electromagnetic system
δi	Current deviation from equilibrium point (A)
δx	Displacement deviation from equilibrium point (m)
$\delta \dot{x}$	Speed deviation from equilibrium point (m/s)
$\delta \ddot{x}$	Acceleration deviation from equilibrium point (m^2/s)
$\delta \psi$	Magnetic flux deviation from equilibrium point (Wb)
$\delta \dot{\psi}$	Voltage deviation from equilibrium point (V)
ζ	Damping ratio
λ	Eigenvalue
μ_c	Core relative material permeability
μ_m	Mass relative material permeability
μ_0	Air permeability (H/m)
ξ_0	Inductance constant
ξ_1	Inductance constant
ξ_2	Inductance constant
ψ	Magnetic flux (Wb)
ψ_0	Magnetic flux at equilibrium (Wb)
$\hat{\psi}_0$	Observer equilibrium point magnetic flux (Wb)
ω	Natural frequency

References

1. Davino, D.; Loschiavo, V.P. Numerical Simulation of Magnetic Materials. In *Reference Module in Materials Science and Materials Engineering*; Elsevier: Amsterdam, The Netherlands, 2020.
2. Schroedter, R.; Csencsics, E.; Schitter, G. Towards the Analogy of Electrostatic and Electromagnetic Transducers. *IFAC-PapersOnLine* **2020**, *53*, 8941–8946. [CrossRef]
3. Ozbay, H. *Introduction to Feedback Control Theory*, 1st ed.; CRC Press: Boca Raton, FL, USA, 1999.
4. Nebyloz, A. *Aerospace Sensors*; Momentum Press: New York, NY, USA, 2012.
5. Erjavec, J.; Thompson, R. *Automotive Technology: A Systems Approach*, 6th ed.; Delmar Cengage Learning: Clifton Park, NY, USA, 2014.
6. Vera Lucia Da Silveira Nantes Button. *Principles of Measurement and Transduction of Biomedical Variables*; Academic Press: Cambridge, MA, USA, 2015.
7. Xiros, N.I. *Feedback Linearization Systems with Modulated States for Harnessing Water Wave Power. Synthesis Lectures on Ocean Systems Engineering*; Morgan & Claypool Publishers: San Rafael, CA, USA, 2020.
8. Regtien, P.; Dertien, E. Inductive and Magnetic Sensors. In *Sensors for Mechatronics*, 2nd ed.; Elsevier: Amsterdam, The Netherlands, 2018.
9. Farmakopoulos, M.G.; Loghis, E.; Nikolakopoulos, P.G.; Xiros, N.I.; Papadopoulos, C. Modeling and Control of the Electrical Actuation System of a Magnetic Hydrodynamic Bearing. In Proceedings of the ASME 2014 International Mechanical Engineering Congress & Exposition IMECE2014, Montreal, QC, Canada, 14–20 November 2014.

10. Bullo, F.; Cortes, J.; Martinez, S. *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Co-Ordination Algorithms*; Princeton University Press: Princeton, NJ, USA, 2009.
11. Tan, Z.; Yang, P.; Nehorai, A. An optimal and distributed demand response strategy with electric vehicles in the smartgrid. *IEEE Trans. Smart Grid* **2014**, *5*, 861–869. [CrossRef]
12. Ogren, P.; Fiorelli, E.; Leonard, N.E. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Trans. Autom. Control.* **2004**, *49*, 1292–1302. [CrossRef]
13. Sun, Q.; Yu, W.; Kochurov, N.; Hao, Q.; Hu, F. A Multi-Agent-Based Intelligent Sensor and Actuator Network Design for Smart House and Home Automation. *J. Sens. Actuator Netw.* **2013**, *2*, 557–588. [CrossRef]
14. Xiros, N.I.; Loghis, E.K. Dynamic Modeling and Identification of a Surface Marine Vehicle for Autonomous Swarm Applications. In Proceedings of the ASME 2012 International Mechanical Engineering Congress & Exposition IMECE2012, Houston, TX, USA, 9–15 November 2012.
15. Xiros, N.I.; Loghis, E. System Identification Using Neural Nets for Dynamic Modeling of a Surface Marine Vehicle. In Proceedings of the ASME 2014 International Mechanical Engineering Congress & Exposition IMECE2014, Montreal, QC, Canada, 14–20 November 2014.
16. Xiros, N.I.; Loghis, E. A Dynamic Interaction Simulator for Studying Macroscopic Swarm Self-organization of Autonomous Surface Watercraft. In Proceedings of the Twenty-Sixth International Offshore and Polar Engineering Conference, Rhodes, Greece, 26 June–1 July 2016.
17. Hu, W.; Liu, L.; Feng, G. Consensus of linear multi-agent systems by distributed event-triggered strategy. *IEEE Trans. Cybern.* **2016**, *46*, 148–157. [CrossRef] [PubMed]
18. Li, X.; Chen, M.Z.Q.; Su, H.; Li, C. Consensus networks with switching topology and time-delays over finite fields. *Automatica* **2016**, *68*, 39–43. [CrossRef]
19. Li, Z.; Ren, W.; Liu, X.; Fu, M. Consensus of multi-agent systems with general linear and Lipschitz nonlinear dynamics using distributed adaptive protocols. *IEEE Trans. Autom. Control.* **2013**, *58*, 1786–1791. [CrossRef]
20. Meng, M.; Liu, L.; Feng, G. Adaptive output regulation of heterogeneous multiagent systems under Markovian switching topologies. *IEEE Trans. Cybern.* **2018**, *48*, 2962–2971. [CrossRef] [PubMed]
21. Olfati-Saber, R.; Murray, R.M. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. Autom. Control.* **2004**, *49*, 1520–1533. [CrossRef]
22. Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and cooperation in networked multiagent systems. *Proc. IEEE* **2007**, *95*, 215–233. [CrossRef]
23. Islam, S.; Xiros, N.I. Robust Asymptotic and Finite-time Tracking for Second-order Nonlinear Multi-agent Autonomous Systems, International. *J. Control. Autom. Syst.* **2019**, *17*, 3069–3078. [CrossRef]
24. Xiros, N.I. Feedback Linearization of Dynamical Systems with Modulated States for Harnessing Water Wave Power. *Synth. Lect. Ocean. Syst. Eng.* **2020**, *1*, 1–75. [CrossRef]
25. Xiros, N.I. Nonlinear Dynamic Analysis for Control of Electromechanical Systems with Coupled Oscillators. *J. Mechatron.* **2015**, *3*, 126–141. [CrossRef]
26. Panofsky, W.K.H.; Phillips, M. *Classical Electricity and Magnetism*; Addison-Wesley Pub. Co.: Reading, MA, USA; New Delhi, Indian, 1962.
27. *MATLAB*; 9.7.0.1190202 (R2019b); The MathWorks Inc.: Natick, MA, USA, 2018.
28. Cairano, S.D.; Bemporad, A.; Kolmanovsky, I.V.; Hrovat, D. Model predictive control of magnetically actuated mass spring dampers for automotive applications. *Int. J. Control.* **2007**, *80*, 1701–1716. [CrossRef]
29. Galushkin, A.I. *Neural Network Theory*; Springer: New York, NY, USA, 2007.