

# Neural Network's Implementation to Control An Active Magnetic Bearing

Roger Achkar and Chaiban Nasr

*Lebanese University Faculty of Engineering I  
Electrical Department  
El Qobbeh, Tripoli, Lebanon  
relachka@hds.utc.fr, chnasr@ieee.org*

Jerôme De Miras and Ali Charara

*Université de Technologie de Compiègne  
Heudiasyc Laboratory (UMR CNRS 6599)  
BP 20529, 60205 Compiègne cedex, France  
demiras@hds.utc.fr, acharara@hds.utc.fr*

## 1 Introduction

The manufacturers and the users of industrial revolving machines are often confronted with technical problems such as heating, frictions, vibrations, maintenance, pollution... moreover, the research tasks showed the interest to increase speeds of machining of materials what requires adapted equipment. The active magnetic bearing (AMB) presents a solution to these problems since it ensures the total levitation of a body in space by electromagnetic forces, thus eliminating any mechanical contact between the rotor and the stator [1]. The development of a new nonlinear model which considers the unbalance (disturbance due to the non-coincidence of the geometric axis and the axis of inertia of the rotor (Fig.1)) presented this disturbance like an intrinsic variable at the system and either like an external additive disturbance [2]. A control law based on the passivity, was developed; as well as other types of control: sliding mode, input-output linearization, and fuzzy controller [3]. Some of these techniques were tested in real time on an active magnetic bearing which turns to 30 000 rd/min, at the Heudiasyc laboratory of the UTC [4].

The goal of our work is to show that the control of the AMB by multilayer perceptrons (MLP) involves an improvement of the response compared to the control of the AMB by classical controllers.

We have recourse to a MLP controller each time the modeling of the system in question is difficult and includes approximations [5],[6]. The MLP have the advantage of learning, i.e. to adapt to new situations even if these situations are not learned with the network during the training phase [7], [8]. In the first time a controller by artificial neural network MLP was developed to control the pin. Several diagrams were developed. The last diagram that was used has all the parameters optimized in order to obtain better results concerning the temporal answers of the positions of the axes [9], [10].

## 2 Active Magnetic Bearing

There are two families of magnetic bearing: Passive magnetic bearing and active magnetic bearing:

The passive magnetic bearing contains a permanent magnet. In this type of MB, we can't modulate the current vector to control the electromagnetic forces.

In the active magnetic bearing AMB, we can modulate the current vector to control the electromagnetic forces [1].

The AMB, presented at the Heudiasyc laboratory of the UTC, is formed by two plane of control. (Y1, Z1) plane and (Y2, Z2) plane. An X axis in the middle and an asynchronous motor (Fig.1).

In order to control the system, a state-input linearization was made [4]. The diagram of the closed loop system is presented (Fig.2). The goal of our work is to show that the control of the AMB by MLP involves an improvement of the response compared to the control of the AMB by classical controllers [10].

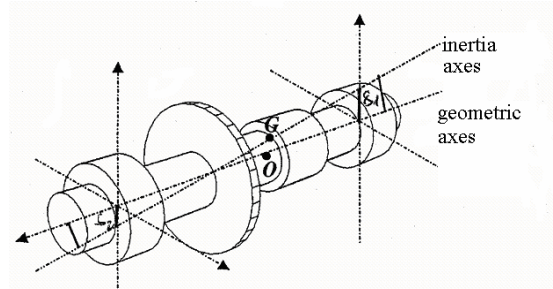


Fig.1. representation of the AMB with an asynchronous motor

Three models are highlighted: A very complicated model in which all the nonlinearity is included, i.e. model very close to the behaviour of the real system. A model including some simplifications, i.e. model of analysis in which the effect of the unbalance is made on one plan of control and a simplified model [11].

TABLE.1  
MECANICAL AND ELECTRICAL PARAMETERS OF THE  
AMB

	Mechanical	parameters
m	3.097 kg	Weight of the rotor
I <sub>x</sub>	8.589x10 <sup>-4</sup> kg.m <sup>2</sup>	Inertia moment for the x axis
I <sub>y</sub>	2.146x10 <sup>-2</sup> kg.m <sup>2</sup>	Inertia moment for the y axis
I <sub>z</sub>	2.146x10 <sup>-2</sup> kg.m <sup>2</sup>	Inertia moment for the z axis
e <sub>0</sub>	0.4 x10 <sup>-3</sup> m	
l <sub>c</sub>	0.1315m	Distance between geometric center and sensors
d <sub>ca</sub>	1.8 x10 <sup>-2</sup> m	
f <sub>v</sub>	1000Hz	Frequency of the first flexible mode of the rotor
Φ <sup>'</sup> <sub>max</sub>	30000tr/min	Rotation maximal speed
	Electrical	parameters
R	0.2Ω	Resistance
L	3mH	Inductance
λ	1.2x10 <sup>-6</sup> mH.m	
R <sub>x</sub>	1.6 Ω	Resistance in relation with the x axis
L <sub>x</sub>	5.8mH	Inductance in relation with the x axis
λ <sub>x</sub>	2.32x10 <sup>-6</sup> mH.m	λ in relation with the x axis
I <sub>max</sub>	6 A	Maximal current
V <sub>max</sub>	50 V	Maximal tension
	Sensors and input	parameters
inputs	-10, +10V for a rotor displacement	-0.25x10 <sup>-3</sup> , +0.25x10 <sup>-3</sup> m
Courant inputs	0, +10V for a current variation	0 : 6A
Tension inputs	0, +10V for a tension variation	-50 : +50V

## 2.1 Equations of the system

The mechanical and the electrical parameters of the AMB are presented in table 1. To define the equations of the model we must do different calculations.

1- The mechanical energy represented by (1) divided into translation kinetic energy and rotation kinetic energy. The calculation is made by considering a moving body in space and the energy must be calculated in function of the positions measured by the sensors.

$$T_m = \frac{1}{2} (m |V_g|^2 + I_x \phi^2) \quad (1)$$

2- The electrical energy and the mechanical potential energy, which indicates the effect of the gravity.

3- The dissipated electrical energy and Deduce the D matrix from (2)

$$D(q) \cdot \ddot{q} + C(q, \dot{q}) \cdot \dot{q} + \frac{\partial \mathcal{R}}{\partial q} = B \cdot F \quad (2)$$

4- Deduce the C matrix from the D matrix by using equation (3)

$$c_k = \sum_{i=1}^6 \sum_{j=1}^6 \frac{1}{2} \left( \frac{\partial d_{kj}}{\partial q_i} + \frac{\partial d_{ki}}{\partial q_j} - \frac{\partial d_{ij}}{\partial q_k} \right) \dot{q}_i \dot{q}_j \quad (3)$$

The equation (3) will be applied for c<sub>1</sub>, c<sub>2</sub>, c<sub>3</sub>, c<sub>4</sub>, c<sub>5</sub>, c<sub>6</sub>. So, the model contains six mechanical equations and ten electrical equations.

The principle is to calculate, by using the D and the C matrix, the outputs of the model represented by X=[x, y1, z1, y2, z2,]', to be able to integrate it into the buckled system [12]. The equations become more complicated. All the equations are written on MATLAB. Simulating system that considers the complicated model will give a real representation of the behaviour of the system controlled by MLP in real time [13], [14].

## 3 Optimal diagram

### 3.1 The first diagrams

We add the MLP bloc to our system in order to improve the temporal responses of the position of the axes on the overshoot, time boarding and response time level. In addition to the implementation of the system, the control by MLP requires two more steps: Learning stage and Online treatment.

In the first diagram the MLP was added to the system controlled by PID by taking the desired positions vector q<sub>d</sub> as input and the φ vector as output added to the error calculated in (4). q<sub>s</sub> Represents the output vector

$$\vec{e} = \vec{q}_d - \vec{q}_s + \vec{\phi} \quad (4)$$

By this emplacement, the MLP tend to minimize the VI vector represented by (5).

$$\vec{V}_1 = \vec{V} - k_p \cdot \vec{\phi} - k_v \cdot \dot{\vec{\phi}} - k_i \int \vec{\phi} dt \quad (5)$$

The simulation on MATLAB showed that the system controlled by this structure of MLP doesn't represent any amelioration on the overshoot, boarding time and response time level for the output responses compared to the system controlled by PID alone. This leads to the necessity of a new structure of MLP to control this system. So, many diagrams were developed.

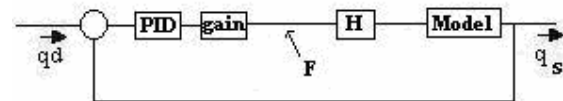


Fig.2. diagram of the closed loop system without MLP

In the second diagram, the output of the system is considered as input to the MLP and its output was added to the error. Simulations made showed that the response time of the system controlled with MLP is much bigger than the one controlled with PID alone. This structure was rejected.

The third diagram consists in replacing the PID controller by a PD controller in order to improve the response time of the system, the output responses diverge.

Another diagram was proposed by taking the output of the system and its derivation as input and we add the output to the error calculated. In this case the MLP seems learning but the responses don't represent a big improvement on the overshoot, time boarding and response time level. Also this structure was rejected.

### 3.2 Optimal diagram

The work done before leads us to an optimal diagram.

We add the MLP to our system by taking the current vector like entry of the network and the vector force like exit (Fig.3).

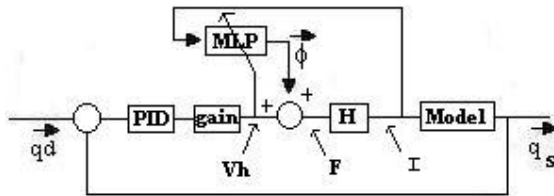


Fig.3 optimal diagram of the closed loop system  
The gain bloc represents the weight of the rotor.

### 3.3 Back-propagation Method

The control of the system by MLP requires two steps: Learning stage and online treatment.

For the optimal diagram obtained before, a back-propagation method was adopted to learn the MLP.

The learning of the MLP is made on line for the two stages. In each stage, direct and back-propagation calculations are made.

**Direct passage:** In the direct passage, we calculate the output of the network by multiplying the weight vector by the input vector. We will consider a network for one hidden layer.

**Back-propagation passage:** this method consists in calculating in a reverse way in order to adjust the weight vector. After calculating in a direct way, we calculate the local gradient in order to adjust the weight vector in each period.

## 4 Simulations and Results

We add the MLP to our system by taking the current vector like entry of the network and the vector force like exit. (Fig.3)

Working with a complicated system that considers the unbalance on two plan of control will be closer to the reality of the machine. The simplifications used in this model are:  $\cos \theta = 1$  and  $\cos \psi = 1$ ,  $\sin \theta = \theta$  and  $\sin \psi = \psi$ ; Centre of gravity of the rotor is equal distance from the 2 plans of control, which implies  $\delta x = 0$ ;  $\delta z$  and  $\delta y$  represent the parameters influencing the existence of the unbalance.

NB:  $\theta$  represents the swing angle around the y axis,  $\psi$  represents the swing angle around the z axis and  $\Phi$  the swing angle around the x axis.

The existence of  $\delta z$  will influence the positions  $z1$  and  $z2$ . The existence of  $\delta y$  will influence the positions  $y1$  and  $y2$ .

For a number of neurons equal to eight in the hidden layer, the learning parameter  $\eta = 0.001$ , the momentum term  $\alpha = 0.1$ , the number of epochs = 15 and the number of examples = 501, a compromise is made between the overshoot, time boarding and response time of the temporal answer. For these optimisations and for  $\delta z = \delta y = 8 \cdot 10^{-5}$  m, a comparison is made between the complicated system with PID alone and the system with MLP.

One can deduce that by increasing the parameters relating to the disturbance, the non linearity becomes more significant, the learned system with MLP has a capacity to reduce this disturbance in order to stabilize the system around the desired answers. Some remarkable oscillations can be eliminated by varying the PID parameters [10].

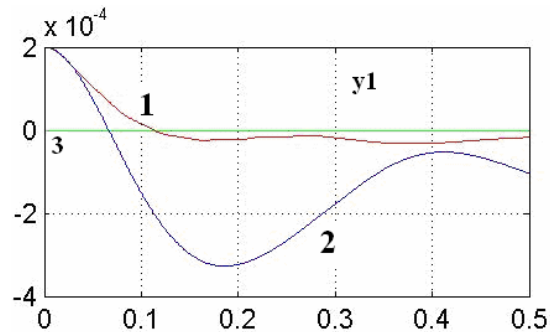


Fig.4: Temporal response of the Y1 axes.  
In red-1: the curve for  $\delta z = 8 \cdot 10^{-5}$  m with MLP, in blue- 2: the curve without MLP and in green-3: the desired answer.

Figure 4 represents the temporal response of the Y1 exit. Y2 exit has the same results. The improvement of the system on the Y1 exit with MLP is very important on the overshoot, response time and convergence of the temporal answer compared to the system controlled by PID alone.

Figure 5 represents the temporal response of the Z1 exit. Z2 exit has the same results. The response is presents many oscillations compared to the one controlled by PID alone.

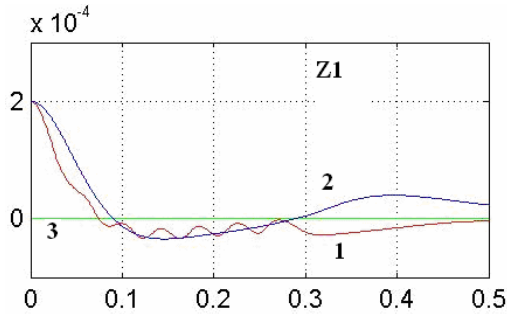


Fig.5: Temporal response of the Z1 axes. In red-1: the curve for  $\delta z = 8 \cdot 10^{-5}$  m with MLP, in blue- 2: the curve without MLP and in green-3: the desired answer.

Figure 6 represents the temporal response of the X exit. The response presents many oscillations but the response is more convergent from the response of the system controlled by PID alone.

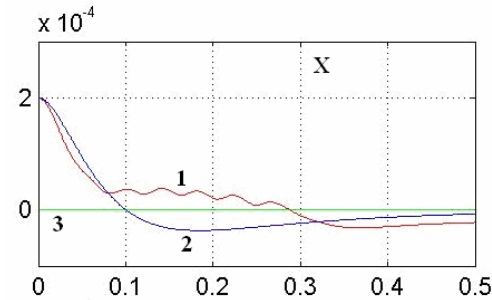


Fig.6: Temporal response of the X axes. In red-1: the curve for  $\delta z = 8 \cdot 10^{-5}$  m with MLP, in blue- 2: the curve without MLP and in green-3: the desired answer.

To improve the convergence of the temporal responses and to reduce the oscillations, variations of the PID parameters are made. Table.2 represents the values of the parameters  $K_p$ ,  $K_d$  and  $K_i$  for the five exits.

TABLE.2  
PARAMETERS OF THE PID

	X	Y1	Z1	Y2	Z2
$K_p$	1500	9600	9300	9600	9300
$K_d$	64	640	640	640	640
$K_i$	2000	56000	55000	56000	55000

A simulation giving the temporal response of the position of the axes by compromising the values of  $K_p$ ,  $K_d$  and  $K_i$  is made. Figure 7 and 8 represent the results of these variations on the Y1 and Z1 outputs. Y2 and Z2 are similar and the X axis doesn't represent any changes.

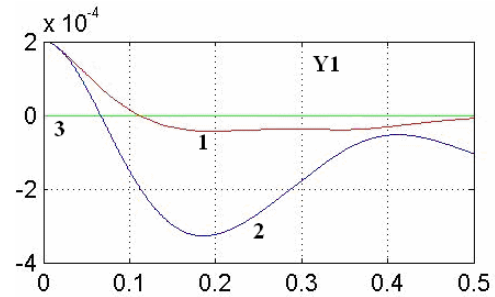


Fig.7: Temporal response on the Y1 axes with a varied PID Parameters. In red-1: the curve for  $\delta z = 8 \cdot 10^{-5}$  m with MLP, in blue- 2: the curve without MLP and in green-3: the desired answer.

By comparing the results obtained for a system with MLP and a system with PID alone, we can deduce that the improvement is acceptable. Even so, it will be very important to improve these results on the overshoot and response time level and reduce the oscillations obtained.

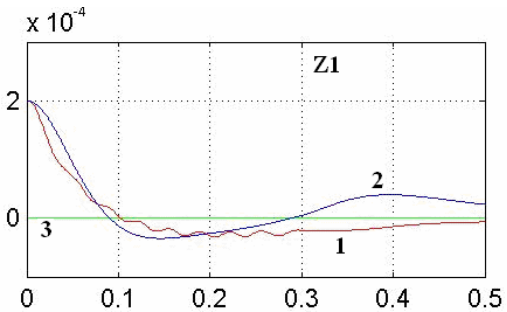


Fig.8: Temporal response on the Z1 axes with a varied PID Parameters. In red-1: the curve for  $\delta z = 8 \cdot 10^{-5}$  m with MLP, in blue- 2: the curve without MLP and in green-3: the desired answer.

## 5 A new method of control

The Optimal diagram used before consists in adding the output of the MLP to the Force vector in order to calculate the current vector.

The temporal response of the position of the axes was improved by varying the parameters of the PID controller. So we will consider an artificial neural network in which we use the output of the network as input of the PID controller.

We add the MLP to our system by taking the current vector like entry of the network and the input of the PID controller like exit. (Fig.9)

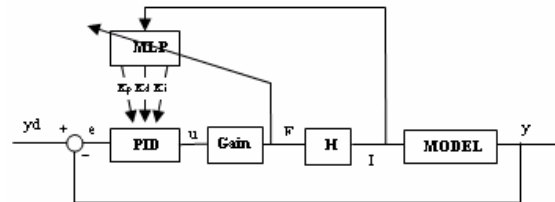


Fig.9. diagram of the closed loop system with an MLP as input to the PID controller.

We control the MLP by calculating the input of the PID controller in order to minimise the error represented by:  $e(t) = y_d(t) - y(t)$ .

## 5.1 Indirect Learning with Back-propagation Algorithm

The basic back-propagation algorithm is based on minimizing the error represented by  $e(t) = y_d(t) - y(t)$ .

Figure 7 represents the artificial neural network tuner.

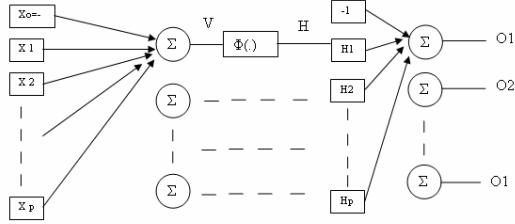


Fig.7. Artificial neural network tuner.

We have to recalculate the entire back-propagation algorithm for both the output and the hidden layer of the neural network. The idea is to update the value of the neural network weights in order to minimize the global error:

$$\xi(t) = \frac{1}{2} \cdot e(t)^2$$

These updating equations are used in the learning phase of the network. The back-propagation phase. Referencing to figure 7, we consider the following parameters:

- $W1$  and  $W2$  are the weights of the hidden and the output layers.
- $H_j$  is the output of the hidden layer.
- $O_k$  is the output of the output layer.

For the direct calculation, the equations are:

$$V_j = \sum \omega_{1ij} \cdot x_i$$

$$H_j = \phi(V_j)$$

$$O_k = \sum \omega_{2jk} \cdot H_j$$

### 5.1.1 Calculation of the output layer

The global error is equal to:

$$e(t) = y_d(t) - y(t)$$

$$e(t)^2 = (y_d(t) - y(t))^2$$

$$\xi(t) = \frac{1}{2} \cdot e(t)^2$$

The minimisation of the global error leads to the minimisation of:

$$\frac{\partial \xi}{\partial \omega_2} = \sum_{k=1}^{15} \left( \frac{\partial \xi}{\partial e} \cdot \frac{\partial e}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial O_k} \cdot \frac{\partial O_k}{\partial \omega_2} \right)$$

$$\bullet \frac{\partial \xi}{\partial e} = e(t)$$

$$\bullet \frac{\partial e}{\partial y} = -1$$

$$\bullet y = Model \times I = Model \times H \times F = Model \times H \times M \times u$$

$$\frac{\partial y}{\partial u} = Model \times H \times M$$

The Model, H and M are represented in the closed loop diagram of figure 9.

We must calculate in what follows the term

$$\frac{\partial u}{\partial O_k}$$

Due to the PID controller, we know that:

$$u = k_p(e(t) + \frac{1}{k_i} \int_0^t e(t)dt + k_d \frac{d}{dt})$$

$$u = k_p(e(t) + \frac{1}{k_i} \times \frac{T_s}{2}(e(t-T_s) + e(t)) + k_d \frac{(e(t) - e(t-T_s))}{T_s})$$

$$u = k_p(e(t) + \frac{1}{k_i} \times \frac{T_s}{2}(z^{-1}e(t) + e(t)) + k_d \frac{(e(t) - z^{-1}e(t))}{T_s})$$

$$u = k_p(1 + \frac{1}{k_i} \times \frac{T_s}{2}(z^{-1} + 1) + k_d \frac{(1 - z^{-1})}{T_s})e(t)$$

A- If the output is  $K_p$ , i.e  $K_{px}, K_{py1}, K_{pz1}, K_{py2}, K_{pz2}$

$$\bullet \frac{\partial u}{\partial O_k} = (1 + \frac{1}{k_i} \times \frac{T_s}{2}(z^{-1} + 1) + k_d \frac{(1 - z^{-1})}{T_s})e(t)$$

B- If the output is  $K_i$ , i.e  $K_{ix}, K_{iy1}, K_{iz1}, K_{iy2}, K_{iz2}$

$$\bullet \frac{\partial u}{\partial O_k} = -\frac{k_p}{k_i^2} \times \frac{T_s}{2}(z^{-1} + 1)e(t)$$

C- If the output is  $K_d$ , i.e  $K_{dx}, K_{dy1}, K_{dz1}, K_{dy2}, K_{dz2}$

$$\bullet \frac{\partial u}{\partial O_k} = \frac{k_p}{T_s}(1 - z^{-1})e(t)$$

And the last term

$$\bullet \frac{\partial O_k}{\partial \omega_2} = H_j$$

The updating equation for the weights of the output layer is:

$$\Delta \omega_{2jk}(t) = \eta \frac{\partial \xi}{\partial \omega_{2jk}} + \alpha \Delta \omega_{2jk}(t-1)$$

### 5.1.2 Calculation of the hidden layer

The same calculations could be done for updating the weights of the hidden layer:

$$\frac{\partial \xi}{\partial \omega_1} = \sum_{k=1}^{15} \left( \frac{\partial \xi}{\partial e} \cdot \frac{\partial e}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial O_k} \cdot \frac{\partial O_k}{\partial H_j} \cdot \frac{\partial H_j}{\partial \omega_1} \right)$$

- $\frac{\partial \xi}{\partial e} = e(t)$
- $\frac{\partial e}{\partial y} = -1$
- $y = Model \times I = Model \times H \times F = Model \times H \times M \times u$

$$\frac{\partial y}{\partial u} = Model \times H \times M$$

- $\frac{\partial u}{\partial O_k}$  the same calculation as for output layer.

$$\frac{\partial H_j}{\partial \omega_1} = \frac{\partial H_j}{\partial V_j} \times \frac{\partial V_j}{\partial \omega_1} = \phi'(V_j) \times x_j$$

The updating equation for the weights of the hidden layer is:

$$\Delta \omega_{1jk}(t) = \eta \frac{\partial \xi}{\partial \omega_{1jk}} + \alpha \Delta \omega_{1jk}(t-1)$$

## 6 Simulation of the new back-propagation method

All these equations are written on Matlab in order to be simulated in the near futur. This method consists in calculating in each period the inputs of the PID controller.

The MLP tends to update his weights on each period in order to minimise the global error represented before. This tuning of the parameters of the PID controller by neural network will improve the temporal response of the position of the axes compared to a system controlled by PID alone.

## 7 Conclusion and perspectives

Simulations done on Matlab show very clearly the improvement of the answers by using the old MLP

method. Using the back-propagation method for an optimal diagram improves the responses obtained by a well learned MLP. The next step we are working on is the real time application. Implementation on DSP of the neural network and studying the behaviour of all process. So we will present in the near future, the results tested on the active magnetic bearing at the Heudiasyc laboratory of the UTC.

## 8 References

- [1] B. Jackson and Y. Destombes. "The potential application of the active magnetic bearings to high power electrical machines," Lse Eng.Bull., vol.15, no. 2, pp. 5-9, 1985
- [2] F.Matsumura and T. Yoshimoto, "System modeling and control design of horizontal shaft magnetic bearing system," IEEE Trans. Magn., vol. Mag-22, May 1986
- [3] K.Nonami and H. Yamaguchi, "Robust control of magnetic bearing systems by means of sliding mode control," in Proc. Int. Symp. Applicat. Electromagn. Forces-ISEM'91, Sendai, Japan, Jan. 1991.
- [4] J. De Miras, A. Charara and B. Caron "Nonlinear control of a magnetic levitation system without premagnetization," IEEE Transactions on Control Systems Technology, September 1996, Vol.4, No.5, pp.513-523.
- [5] K. Warwick, G. W. Irwin, and K. J. Hunt (eds.), Neural Networks for control and systems, Peter Peregrinus Ltd (1992).
- [6] K. Warwick, Neural networks for control: counter arguments, Proc. IEE INT. Conference Control 94, Warwick University, pp.95-99 (1994).
- [7] A. H. Slocum, Precision Machine Design. Englewood Cliffs, NJ: Prentice-Hall, 1992, pp. 625-639
- [8] K. S. Narendra and K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Trans. on neural networks, 1, pp.4-27 (1990)
- [9] R. Achkar and C. Nasr, Non Linear Control of an Active Magnetic Bearing, DEA memory, Lebanese University, September 2003.
- [10] R. Achkar, C. Nasr and A. Charara, "Control of an Active Magnetic Bearing Using MLP", Conf in the Artificial Neural Networks In Eng, Missouri, USA, 7-9 November 2005, pp. 811-820
- [11] J. Salm and G.Schweitzer, "Modeling and control of a flexible rotor with magnetic bearings," in Proc. 3<sup>rd</sup> Int. Conf. Vibrations in rotating machinery, Univ. York, U.K., Sept. 11-13, pp. C277-C284, 1984
- [12] K.S.Narendra and K. Parthasarathy, Gradient methods for the optimization of dynamical systems containing neural networks, IEEE Trans. on Neural Networks, 2, pp.252-262, 1991
- [13] A. Charara, B. Caron, and G. Lemarquand, "Modeling and non-interactive control of an active magnetic bearing," Int. J. Appl. Electromagnetic. In materials, no. 2, pp. 359-368, 1992
- [14] A. Charara and B. Caron, "Magnetic bearing: Comparison between linear and nonlinear functioning," in Proc. 3<sup>rd</sup> Int. Symp. Magnetic Bearings, pp. 451-463, 1992