

Artificial Neural Network vs PID Controller for Magnetic Levitation System

G.M.K.B. Karunasena

Mechanical Engineering Department
The Open University of Sri Lanka
Nawala, Sri Lanka

H.D.N.S. Priyankara

Mechanical Engineering Department
The Open University of Sri Lanka
Nawala, Sri Lanka

B.G.D.A. Madhusanka

Mechanical Engineering Department
The Open University of Sri Lanka
Nawala, Sri Lanka

Abstract:- This research investigates the acceptability of the Artificial Neural Networks (ANN) over the PID Controller for the control of the Magnetic Levitation System (MLS). In the field of advanced control systems, this system identifies as a feedback-controlled, single input- single output (SISO) system. This SISO system used a PID controller for vertical trajectory controlling of a metal sphere in airspace by using an electromagnetic force that directed to upward. The vertical position of the metal sphere controlled according to the applied magnetic force generated by a powerful electromagnet and the electromagnetic force controlled by varying the supply voltage. To control this nonlinear system, we develop a multilayer artificial neural network by using Matlab software and integrate that with the physical magnetic levitation model. According to specific initial conditions, the actual responses of the magnetic levitation system with artificial neural network compares the desire response of the metal sphere. The ability of control this nonlinear system by using neural networks validate by comparing results obtained by the PID controller and artificial neural network.

Keywords:- Magnetic Levitation System, Artificial Neural Networks, PID, Linear Model, Nonlinear Model Control.

I. INTRODUCTION

The Magnetic Levitation System is a basic model of the approach behind magnetic levitation systems such as maglev trains, magnetic suspension systems and magnetic bearings. Maglev trains are being test recently, and certain trains are already available. Magnetic suspension systems are an alternative automobile suspension system that uses magnetically controlled shock absorbers to active suspension. The magnetic levitation systems have additional active vibration damping capabilities. These capabilities can be achieve by using different control algorithms without any mechanical developments of the system [1]. The Maglev systems can control with different controllers and test in real-time using MATLAB Simulink environment. Neural networks can map nonlinear functions flexibly and arbitrarily. Therefore neural networks more suitable for the control of nonlinear systems. Neural networks comprise an incredible data-modelling tool that can capture and represent complex input/output connections. Neural networks use mathematical algorithms to perform but it needs only input and output data for adjusting the system parameters. Depending on the accuracy of the results, a neural network

controller requires more number of layers and neurons. Although PID control is an efficient technique for handling non-linear systems but modelling of PID controller is not a simple case and controller needs suitable PID values to stabilize the system. PID controller use a linearization approach of the dynamics of the systems and reduces the effects of nonlinear facts. Using this approach some systems can be stabilize near their nominal point of operation [2]. Due to that reason, PID controller is not suitable for nonlinear control application. The other important aspect is that the PID control needs the device model for the determination of the parameters of PID controller; sometimes it is hard to develop a mathematical model of the complex systems. Artificial Neural Networks (ANN) are more suited to processes that are too complex to study and build using traditional techniques. In neural networks, the system behavior is implement by control algorithm based on trained neurons. For control adjustments, the neural network requires only modification in these trained neurons. Therefore, control system based on neural networks not only simplifies the design and reduces the task of solving complex mathematical equations of nonlinear systems.

II. MODELING OF MAGNETIC LEVITATION SYSTEM

The Physical setup of the Magnetic levitation system shown in Fig.1. The system's function is to control the ball's vertical location by changing the current through the input voltage in the electromagnet. The main components are an optoelectronic position sensor, electromagnetic actuator coil and a suspended ferromagnetic ball. The controller is implemented in computer software and connect with a physical device through an electronic hardware interface card. The optoelectronic sensor determines the vertical location of the ferromagnetic ball and transfers it to a controller through an interface card with electronic hardware. According to the difference between desired and measure output, the controller sends signal to the current controller of the actuator. The actuator consists of an electromagnet to produce upward attractive force for levitation against gravity on the ferromagnetic ball. This method consists of two stages; In the first stage, the metal sphere transfer to the desired location. In the second stage, the metal sphere keeps the desired position. The required magnetic force for position control of the metal sphere generated by the electromagnet, which controlled by a close loop control system. Therefore this system consists of nonlinear electromechanical equations and this system has nonlinear characteristics [3]. The magnetic levitation system

can be divide into two systems: a mechanical system and an electrical system.



Fig 1:- Magnetic levitation experimental system

A. Mechanical Modeling

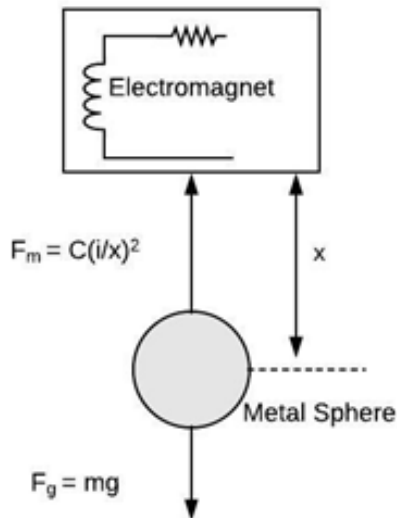


Fig 2:- Free body diagram of metal sphere

The Free body diagram of metal sphere controlled by balancing the electromagnetic force F_m and gravitational force F_g shown in Fig.2. Net force F_{net} acting on the ball is given by Newton’s 3rd law of motion while neglecting friction, drag force etc. Equation (3) shows motion equation of the metal sphere.

$$F_{net} = F_m - F_g \tag{1}$$

$$F_m = C \left(\frac{i}{x} \right)^2 \tag{2}$$

$$m\ddot{x} = mg - C \left(\frac{i}{x} \right)^2 \tag{3}$$

B. Electrical Modeling

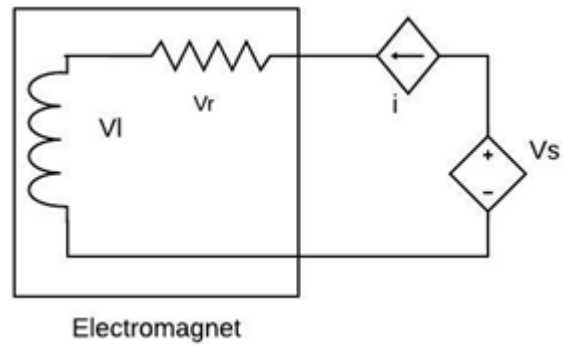


Fig 3:- Circuit Diagram of Electromagnet

The circuit diagram of the electromagnetic system shown in Fig.3. The electromagnetic force produced by a current, given by the Kirchoff’s voltage law (4) while neglecting other internal resistances and inductances.

$$V(t) = V_R + V_L = iR + \frac{dL(x)i}{dt} \tag{4}$$

C. Linear Model Development

Nonlinear model of magnetic levitation system can be described in terms of the following set of differential equations based on electro-mechanical modeling by linearizing of (2).

$$F_m = C \left(\frac{i_0}{x_0} \right)^2 + 2C \left(\frac{i_0}{x_0^2} \right) i(t) - 2C \left(\frac{i_0^2}{x_0^3} \right) x(t) + \dots \tag{5}$$

Neglect the higher order terms, rearrange (3) and substitute (6).

$$m\ddot{x} = mg - C \left(\frac{i_0}{x_0} \right)^2 - 2C \left(\frac{i_0}{x_0^2} \right) i(t) + 2C \left(\frac{i_0^2}{x_0^3} \right) x(t) \tag{6}$$

In (5) and (6), i_0 equals the current of the electromagnet coil when the ball is at x_0 . When the magnetic force equals the downward force on the ball, the acceleration of the ball becomes to zero and (6) becomes,

$$m\ddot{x} = -2C \left(\frac{i_0}{x_0^2} \right) i(t) + 2C \left(\frac{i_0^2}{x_0^3} \right) x(t) \tag{7}$$

By taking Laplace transformation of (7)

$$X_S = I_S \left(\frac{-2C \left(\frac{i_0}{x_0^2} \right)}{mS^2 - 2C \left(\frac{i_0^2}{x_0^3} \right)} \right) \tag{8}$$

By taking Laplace transformation of (4)

$$I_S = \left(\frac{V_S}{LS+R} \right) \tag{9}$$

The displacement of the ball measured by the optical sensor sensor, β is sensor gain.

$$V_x(t) = \beta x(t) \tag{10}$$

By taking Laplace transform of (10)

$$V_x(s) = \beta_{x(s)} X_s \tag{11}$$

By using (8), (9) and (10)

$$\frac{V_x(s)}{V_s} = \left(\frac{-2C\beta \left(\frac{i_0}{mLx_0^2} \right)}{\left(\frac{R}{L} + s \right) \left(s^2 - 2C \left(\frac{i_0^2}{mx_0^3} \right) \right)} \right) \tag{12}$$

Equation (12) is the linear model transfer function of the magnetic levitation model [3].

III. NEURAL NETWORK CONTROLLER DESIGN

This section mainly focused on, design and development of proposed neural network controller for magnetic levitation system.

A. Neural Network Controller

Neural networks concept is an algorithm implementation of the neuronal structure of our brain. Neural network idea began with physiologist's research, Warren McCulloch, and mathematician Walter Pitts when they built a basic neural network using electronic components in 1943 to describe how neurons will function inside the brain [4]. A neuron is the essential processing element of a neural network. A biological neuron takes signals from other sources, integrates them in some way, performs an input operation, which is generally nonlinear and then produces the result. In artificial neural networks, this process artificially done by using artificial neuron. Synaptic weights, summing function and activation function are the main elements of an artificial neuron model. A synapse is a connecting link between two neurons and characterized by a weight known as synaptic weight or interconnection weight. Positive weights correspond to excitatory synapses, while negative weights to inhibitory ones. At the summation point, the summing function sums all the outputs which coming from other branches and add bias value to respective neuron. Then the input values to the neuron map to certain finite value by using the activation function. The structure of an artificial neuron shows in Fig. 4.

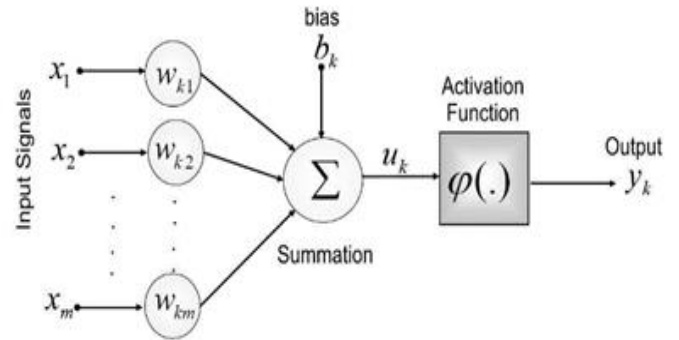


Fig 4:- Structure of Artificial Neuron

For the above general model of artificial neural network, the net input u_k can be calculated as (15).

$$u_k = x_1 W_{k1} + x_2 W_{k2} + \dots \dots \dots + x_m W_{km} \tag{15}$$

The output can calculate by using activation function inside the node and net input used as input for the activation function according to input value the activation function (16) generates the node output.

$$y_k = \varphi (u_k) \tag{16}$$

The modifying process of weights between neurons in a network called the learning process. This learning process can be categorized three types of Supervised Learning, Unsupervised Learning, and Reinforcement Learning [5].

B. Neural Network Controller Development

The neural network control algorithm used for magnetic levitation system can explain as; the magnetic levitation system controlled by using multilayer neural network because multilayer networks are more powerful than single-layer networks. The two-layer network having a hyperbolic tangent sigmoid first layer and a hyperbolic tangent sigmoid second layer trained to approximate most functions arbitrarily well. Single layer networks cannot do this. The proposed multilayer feed-forward neural network consists in one input layer, one output layer and two-hidden-layers, as shown in Fig. 5. In this neural network input layer consists two inputs an output layer consists signal output.

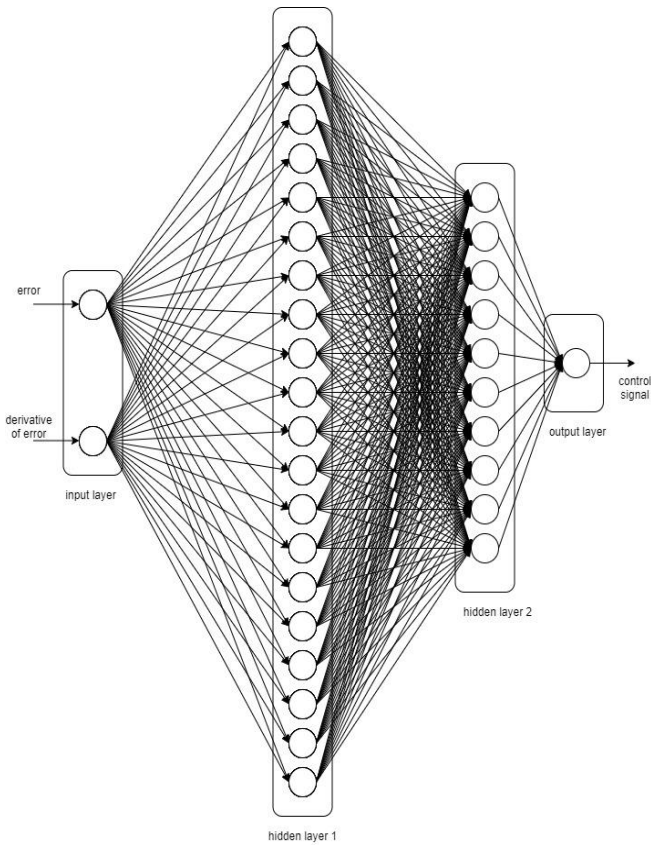


Fig 5:- Multilayer neural network

The difference between actual response and desire response (error) and the derivative of the error de are selected as the inputs for the input layer and the output (control signal) assumed to the reference voltage for current controller. The hidden layer 1 consists 20 neurons as shown in Fig. 6 and hidden layer 2 consists 10 neurons as shown in Fig. 7.

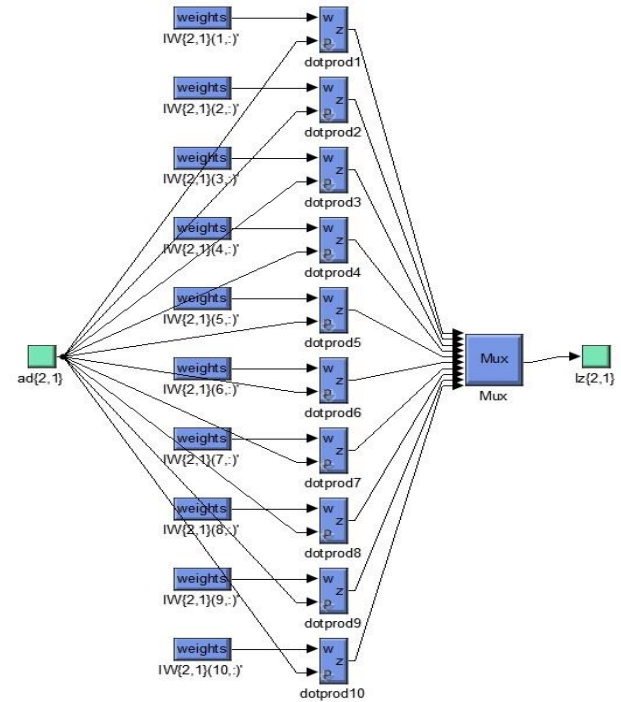


Fig 7:- Matlab Simulink model of hidden layer 2

The activation function use for hidden layer 1 & 2 neurons is hyperbolic tangent sigmoid function, defined in (17). The hyperbolic tangent sigmoid function defined in (17), selected as the activation function for all the hidden layer nodes. This ensures the values stay within a relatively small range and enable the neural network to learn nonlinear relationships of the system.

$$a = \frac{e^n - e^{-n}}{e^n + e^{-n}} \tag{17}$$

The transformation between the input and output of each neuron can range to interval -1 to +1 by using the anti-symmetric function, as shown in Fig. 8.

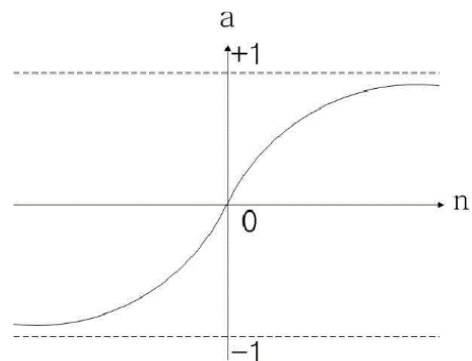


Fig 8:- The hyperbolic tangent sigmoid function

The activation function use for output layer neuron is linear function, defined in (18).

$$a = n \tag{18}$$

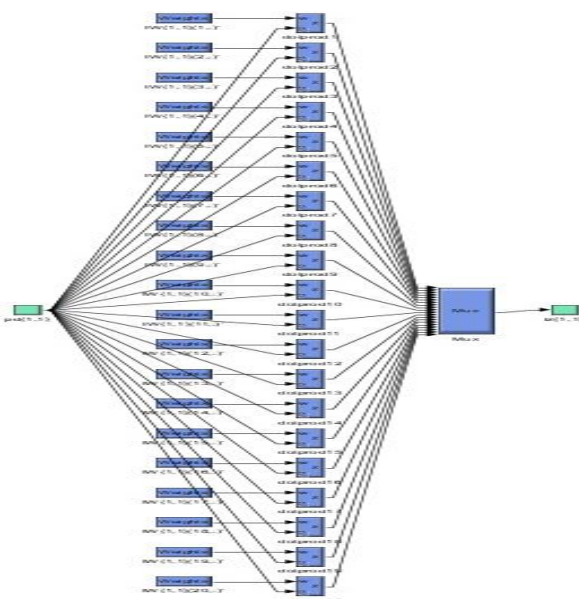


Fig 6:- Matlab Simulink model of hidden layer 1

The input layer consists of the input nodes without input links connected to them and the last layer consist of a single node. The intermediate hidden layers consist of neurons, which connected each other.

C. Neural Network Controller Training

The multilayer neural network trained by using the Backpropagation learning method based on Levenberg Marquardt algorithm. The Backpropagation is a learning method that comes under Supervised Learning. The Backpropagation based on gradient descent method, which decreases the error between inputs and outputs by decreasing the gradient of the error curve [6]. This algorithm trains the multilayer neural network for the set of input samples with known output samples. The training process done by using 30000 samples for inputs and output of the neural network. These samples collected through identification by running the maglev system with PD controller. The neural network trained by using Matlab neural network toolbox shown in Fig. 9.

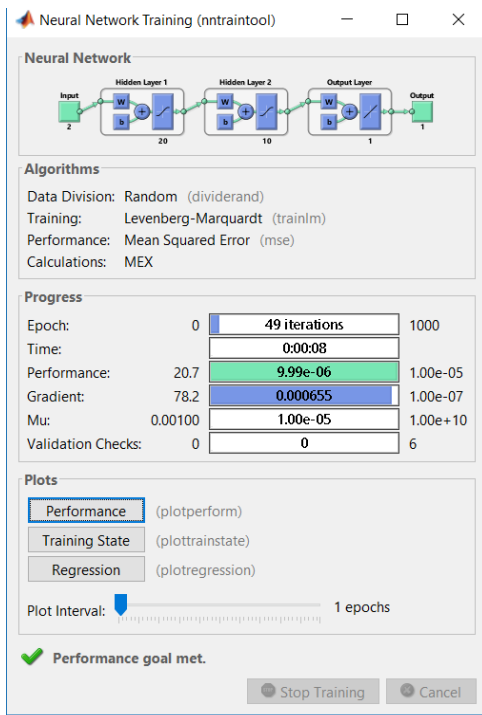


Fig 9:- Training the neural network

By using the Levenberg Marquardt algorithm, we can minimize the sum of the square error function (19). The Levenberg Marquardt algorithm is a combination of two optimization methods; the gradient descent method and the Gauss-Newton method.

$$E = \frac{1}{2} \left(\sum k(e_k)^2 \right) = \frac{1}{2} \|e\|^2 \tag{19}$$

In the equation (19) e_k is denote the error in k^{th} data input and e denote a vector with elements e_k . If the difference between new and previous weight vectors is small then the error vector can express as in (20)

$$e_{(j+1)} = e_{(j)} + \frac{\delta e_k}{\delta w_i (w_{(j+1)} - w_{(j)})} \tag{20}$$

Then, the error function can be express as in (21)

$$E = \frac{1}{2} \left\| e_{(j)} + \frac{\delta e_k}{\delta w_i (w_{(j+1)} - w_{(j)})} \right\|^2 \tag{21}$$

The new weight vector can obtain by minimizing the error function, gives

$$w_{(j+1)} = w_{(j)} - (Z^T Z)^{-1} Z^T e_{(j)} \tag{22}$$

Where,

$$(z)_{ki} = \delta e_k / \delta w_i \tag{23}$$

Since the Hessian for the sum-of-square error function is,

$$\begin{aligned} (H)_{ij} &= \delta^2 E / \delta w_i \delta w_j \\ &= \sum \left\{ \left(\delta e_k / \delta w_i \right) \left(\delta e_k / \delta w_j \right) + e_k \delta^2 e_k / \delta w_i \delta w_j \right\} \end{aligned} \tag{24}$$

Neglecting the second term, the Hessian can written as

$$H = Z^T Z \tag{25}$$

Hessian matrix consists first order derivatives of the network weights that are simply calculate by Backpropagation. The weight updating formula working on weights until the error function get into minimize but this process takes large amount of calculating steps, which tends to invalidation of linear approximation of error function. In the Levenberg-Marquardt algorithm the validity of linear approximation, ensure by using modified error function (26).

$$E = \frac{1}{2} \left\| e_{(j)} + \frac{\delta e_k}{\delta w_i (w_{(j+1)} - w_{(j)})} + \lambda \|w_{(j+1)} - w_{(j)}\|^2 \right\|^2 \tag{26}$$

The λ is a parameter, which control the step size. Minimizing the modified error with respect to $w_{(j+1)}$ gives,

$$w_{(j+1)} = w_{(j)} - ((Z^T Z + \lambda I)^{-1} Z^T e_{(j)}) \tag{27}$$

The gradient decent use large amount of λ values and Newton method use small amount of λ values [7].

IV. INTERFACING ANN CONTROLLER

The default Feedback PID controller replaces by the Neural Network controller as shown in Fig. 10. The Madlab derivative block use to generate derivative values of error after the summing point for the Neural Network input. The output control signal form the Neural Network connected to the Feedback digital to analogue converter. The neural network controller’s inputs were in between -2V and +2V. However, during the stable position, the output from the summing point is very low therefore, the controller not perform properly and the metal sphere moves to an unstable region. To remove this problem, two gains has been used to amplify the values of input and error to match the range of -2V and +2V.

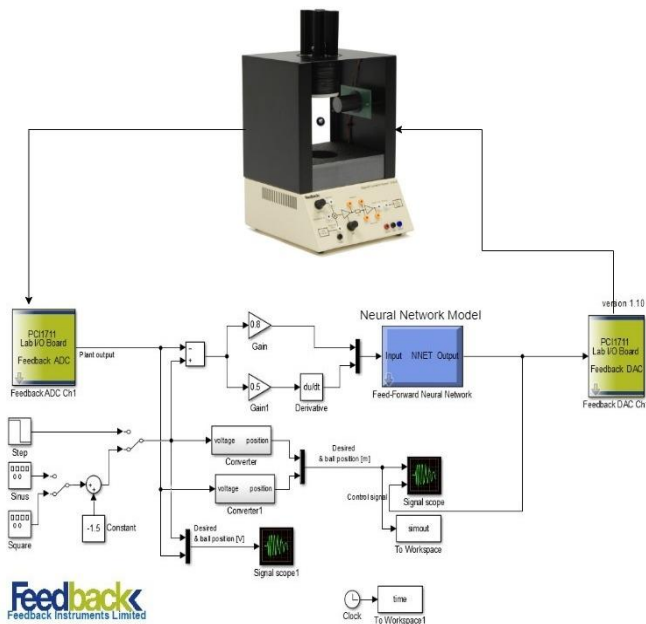


Fig 10:- Interface neural network



Fig 11:- Physical setup

V. RESULTS AND DISCUSSION

The control technique of an artificial neural network controller depends on amount of human experience data, learning algorithm, number of hidden layers. The response of neural network controller is examination for different sorts of inputs in the presence of many unmeasurable disturbances such as air resistance, moisture, change in coil resistance with time due to the generated heat in coil etc. For testing of developed neural network controller, the square wave, sine wave and step input applied neural network controller. The performance of the designed neural network controller was compare and analyze with the professionally tuned PID. We use tuned PID values for the PID controller to control the vertical position of the metal sphere. These PID values recommended by Feedback Instruments. The values are Kp-4, Kd-0.2, and Ki-2. Then, response of system for PID and neural network controller compared for validation. After analyzing results, neural network controller has better tracking response as contrast with the PID controller. The compered results for the step, sine and square inputs shown in Fig. 12, Fig. 13 and Fig.14. In the step response sown in Fig. 11, initial value is 0 and after 3 seconds -1.5V step is given. The comparative data obtained through the analysis are as shown in the Table I.

Parameter	PID controller	ANN controller
% Overshoot	27%	2.8%
Settling time	5.71s	1.12s
Rise time	0.1s	0.46s
Delay time	0.05s	0.36s
Peak time	0.22s	0.48s

Table 1:- Transient analysis between PID and ANN controller

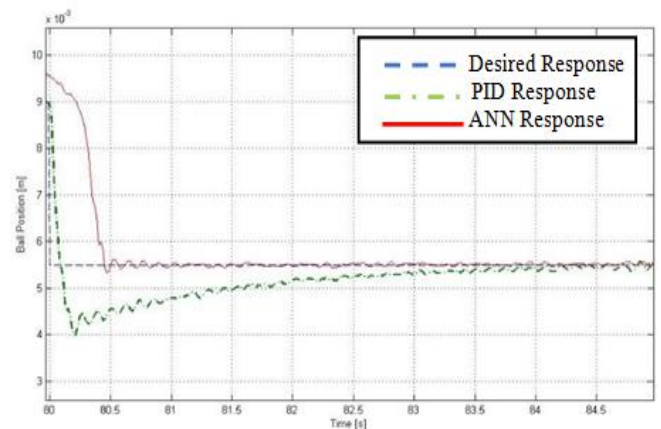


Fig 12:- Step response

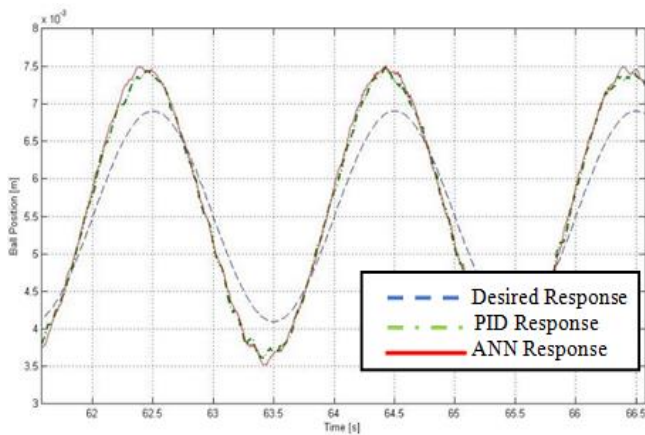


Fig 13:- Sine wave response

Fig. 13 shows the response of the system when a 0.4V amplitude sine wave input. It shows that the ANN controller follow the sine wave trajectory same as PID controller and Fig. 14 shows the response of the system when a square input. This shows the ANN controller closely follow square wave input with lower overshoot than PID controller.

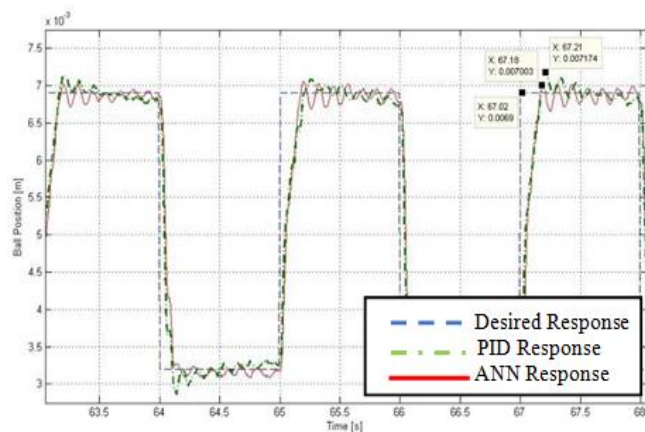


Fig 14:- Square wave response

VI. CONCLUSION

In this research, the Artificial Neural Network controller successfully designed and developed by using Matlab to control the vertical position of metal sphere in airspace. Based on the experimental results, it shows the Artificial Neural Network controller successfully stabilize the system and efficiently follow the desired trajectory same as PID controller response. Due to the lack of overshoot, the neural network response becomes quickly to a stable point. The response time of the neural network controller is much longer than PID controller due to the less overshoot but neural network controller provide better settling time and less overshoot for a step input.

ACKNOWLEDGMENT

The authors would like to thank control systems laboratory at The Open University of Sri Lanka for providing the facilities to conduct this research.

REFERENCES

- [1]. Yadav, Shekhar & J.P.Tiwari, J.P.Tiwari & Nagar, S.K.. (2012). Digital Control of Magnetic Levitation System using Fuzzy Logic Controller. International Journal of Computer Applications.
- [2]. G.M.K.B. Karunasena, H.D.N.S. Priyankara, B.G.D.A. Madhusanka (2019). Design and Development of Fuzzy Logic Controller for Magnetic Levitation System. Conference: International Conference on Artificial Intelligence, SLAAI-ICAI-2019
- [3]. H.D.N.S. Priyankara, Singhe Wijesinghe, T.M.W. Vithanawasam (2018). Fuzzy Logic Controller vs PID Controller for Real Time Magnetic Levitation System. Conference: International Conference on Information and Automation for Sustainability, ICIAFS 2018
- [4]. Piccinini, Gualtiero. (2004). The First Computational Theory of Mind and Brain: A Close Look at McCulloch and Pitts' "Logical Calculus of Ideas Immanent in Nervous Activity". Synthese. 141. 10.1023/B:SYNT.0000043018.52445.3e.
- [5]. Sathya, R. & Abraham, Annamma. (2013). Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification. International Journal of Advanced Research in Artificial Intelligence. 2. 10.14569/IJARAI.2013.020206
- [6]. Singh, Sapna. (2012). Backpropagation Learning Algorithm Based on Levenberg Marquardt Algorithm. Computer Science & Information Techno.
- [7]. Colak, İlhami & Sahin, Murat & Esen, Zafer. (2013). Artificial Neural Networks Controller Algorithm Developed for a Brushless DC Motor.