Eng.& Tech. Journal ,Vol. 28, No.19,2010

Speed Control of Permanent Magnet D.C. Motor Using Neural Network Control

Lina J. Rashad*

Received on: 9/2/2010
Accepted on: 30/6/2010

Abstract

This paper proposes the speed control of a permanent magnet direct current (PMDC) motor by varying armature voltage. The objective is to control the rotor angular speed to follow the desired value. The main feature of the proposed controller is neural network, which captures the nonlinearity system of the motor. Neural network (NN) performance is compared with the conventional controller performance like PI (Proportional-Integral) controller to show that NN performance is excellent. Numerous work reported in recent past have shown that Artificial Neural Network (ANN) controller has a potential to replace the conventional PI controller. Artificial Neural Network control apparently offers a possibility of obtaining an improvement in the quality of the speed response, compared to PI control. This research proposes NARMA-L2 (Nonlinear Autoregressive-Moving Average) as an improved ANNtechnique, and trained as a close loop controller, which gives an ideal performance as compared with PI controller to control the angular speed of rotor in a permanent magnet dc (PMDC) motor. Simulation results show the effectiveness of the proposed control schemeThe entire system has been modeled using MATLAB toolbox.

Keywords: PMDC Motor; speed control of PMDC, PI controller, ANN controller, NARMA-L2 controller.

السيطرة على سرعة المحرك ذو تيار ثابت ذو مجال مغناطيسي دائم باستخدام الشبكة العصبية الاصطناعية

الخلاصة

هذه الدراسة لغرض السيطرة على سرعة محرك ذو تيار مستمر ذو مجال مغناطيسي دائم ذو فولتية محرك مختلفة. إنّ الهدف من هذه الدراسة هو السيطرةَ على السرعةِ الزاوية للمحرك لإتّباع القيمةِ المطلوبةِ .إنّ الميزّةَ الرئيسية لجهازِ السيطرة المُقتَرَح (الشبكة العصبيةِ الاصطناعية) هو استيعاب النظام اللاخطي للمحرك. يُقَارَنُ أداء مسيطر الشبكة العصبية مع أداء مسيطر تقليدي مثل جهاز المسيطر(التناسبي – التكاملي) لبيان أمتياز مسيطرة الشبكة العصبية. ان العديد من البحوث التي صدرت في الماضي الحديث اضهرت ان مسيطرات الشبكات العصبية الأصطناعية حلت محل المسيطر(التناسبي – التكاملي) التقليدي. ان الشبكة العصبية الاصطناعية اظهرت مدى فعاليتها في تحسين أستجابة سرعة المحرك للسيطرة

*Electrical and Electronic Engineering Department, University of Technology/Baghdad

بالمقارنة مع المسيطر (التناسبي – التكاملي). يقدم هذا البحث (NARMA-L2) كتقنية محسنة للشبكات العصبية الاصطناعية والتي تُدَرَّب كمسيطر حلقي مغلق والذي يعطي اداء مثالي بالمقارنة مع المسيطر (التناسبي – التكاملي) للسيطرةعلى سرعة المحرك ذو تيار مستمر من نوع مجال مغناطيسي دائم. تضهر نَتائِجُ المحاكاةِ فعالية مخطط السيطرةِ المُقتَرَح. ان نضام المحاكاة المستخدم هو برنامج ماتلاب.

1. Introduction

Permanent magnet dc motors are useful in a range of applications, from battery powered devices like wheelchairs and power tools, to conveyors and door openers, welding equipment, X-ray and tomographic systems, pumping equipment. They are frequently the best solution to motion control and power transmission applications where compact size, wide operating speed range, ability to adapt to a range of power sources or the safety considerations of low voltage are important and They produce relatively high torques at low speeds, enabling them to be used as substitutes for gearmotors in many instances. Because of their linear speed-torque curve, they particularly suit adjustable speed and servo control applications where the motor will operate at less than 5000 rpm inside these motors, permanent magnets bonded to a flux-return ring replace the stator field windings found in shunt motors. A wound armature and mechanical brush commutation system complete the motor. The permanent magnets supply the surrounding field flux, eliminating The need for external field current. This design yields a smaller, lighter, and energy efficient motor [1, 2].

Conventional controllers like PI require accurate mathematical models describing the model of the system under control. Even if a model can be obtained for the system under control one of the main difficulties with the conventional tracking controllers for electric drives is their inability to capture unknown load

Characteristics over a widely ranging operating point. This makes tuning of respective parameters difficult [3, 4]. Some adaptive control techniques, such as, variable structure control, self–tuning does not need a model for system dynamics. Although these adaptive controllers are effective, their hardware implementation can be elaborated. ANN can be trained to control non–linear plant by presenting a suitable set of input and output data generated by the plant. The NN has several features that make it highly suitable for dc motor applications. For example, a neural network can generate a non–linear mapping between the inputs and outputs of an electric drive system without the need for a predetermined model. Hence neural networks are preferred [5].

In this paper non–linear permanent magnet dc (PMDC) motor is controlled by NNC and compared the obtained results with conventional PI controller for speed control. The results of the simulation are given.

2. Modeling and Simulation of Permanent Magnet D.C. Motor

The Permanent Magnet D.C. Motor model used in this study is shown in fig. (1) [6] [7]. This type of motor uses a permanent magnet to generate the magnetic field in which the armature rotates, no field current, that mean the flux must be constant. The electrical armature and field circuit can model the motor. In this simple model R_a and L_a indicate the equivalent armature coil resistance and inductance respectively and R_F and L_F indicate the equivalent field resistance and inductance respectively , v_a is the voltage supplied by the power source. Kirchoff's voltage law leads to the following equation:

$$v_a = e_m + R_a i_a + L_a (di_a/dt) \quad \text{…. (1)}$$

Where e_m is the back electromotive force. By examining the effect of the magnetic field in the motor, and realizing that magnetic flux is constant, we can arrive at the following two equations relating the electrical torque T_e and the motor speed output ω to the supplied current and voltage:

$$e_m = K \varphi \omega \quad \text{…… (2)}$$

$$T_e = K \varphi i_a \quad \text{….. (3)}$$

where K is the motor constant and it's dependent on the particular motor, but if it is expressed in SI Units, their values are always equal and φ is the flux of motor. In addition to the electrical equations, we should consider the mechanical equation of PMDC motor and load effect. The mechanical equation of PMDC motor is:

$$T_e - T_L = j\frac{dw}{dt} + Bw \quad \text{….. (4)}$$

Where: T_L is the torque of the mechanical load; j is the inertia of motor load and B is the damping coefficient associated with the mechanical rotational system of the motor. The block diagram obtained from these equations of the permanent magnet dc motor is shown in fig.(2) . The used PMDC motor parameters are: 3000 rpm, 220 v., L_a= 3.2 mH, R_a=0.2 Ω, j= 0.15 kg/m^2, B=0.0001N.m. T_L=10 N.m.

The transfer function block diagram of the permanent magnet dc motor without controller can be developed in "Matlab" as shown in fig.(3). This type of PMDC motor consists of variable armature voltage, which is connected directly to PMDC motor, and the operation performance for different speed responses without controller is shown in figure (4). From figure (4) it's seen that the period of simulation in the speed responses are set as 1 seconds, and by studying the behavior of the motor speed without controller , we can see that the settling time and steady-state error are high because the motor is un controlled that mean the operation performance for different speed responses of this motor are very slowly. Therefore, to improve system performance and get exact operating speed, close loop control must be implemented. The following proposed models are needed to study the effect of using the PI controllers and the Neural Network controller (NNC) which represent on the second order model of PMDC motor for speed control .

3. PI controller

PI controller is a common sense approach to control based on the nature of error. It can be applied to wide varieties of systems. PI controller has two parameters, the two parameters that must be determined (some times, must be optimized) for the given process, to give the desirable output responses for the plant are: proportional gain and integral gain[4]. The error signal (e) will be sent to the PI controller, and the controller computes both the proportional and the integral of this error signal. The signal (u) just past the controller is given as:

$$u= K_p.e + K_i \int e.dt \qquad \text{........ (5)}$$

This signal will be sent to the plant, and the new output (y) will be obtained. This new output (y) will be sent back to the sensor again to find the new error signal (e). The controller takes this new error signal and computes its proportional and its integral again. This process will continuous until the desired output achieved. A proportional controller (K_p) will have the effect of reducing the rise time and will reduce, but never eliminate, the steady –state error. An integral control (K_i) will have the effect of eliminating the steady– state error, but it may make the transient response worse [4]. The PI controller is introduced in the PMDC motor to improve the dynamic response and also reduces the steady state error. The block diagram for simulating a permanent magnet of dc motor with PI controller and the operation performance for different speed responses with PI controller are shown in Figures (5) and (6) respectively. It is very interesting to investigate the effects of each of PI controller's parameters K_p and K_i on the PMDC motor speed control. By trial and error the proportional gain (K_p) and integral gain (K_i) selected as (K_p=1, K_i=30). The time taken for the speed response (ω) to reach the desired value (settling time) is now 0.2 seconds. When these results are compared with the same system without controller it's found that these settings of K_p and K_i produce a good overall response.

4. Neural Network controller

Neural networks have been applied very successfully in the identification and control of dynamic systems [8,9]. The universal approximation capabilities of the multilayer perception make it a popular choice for modeling nonlinear systems and for implementing general-purpose nonlinear controllers [9]. There are three popular neural network architectures for prediction and control that have been implemented in the Neural Network Toolbox [10]:

•Model Predictive Control.

•NARMA-L2 (or Feedback Linearization) Control.

•Model Reference Control.

NARMA-L2 (Nonlinear Autoregressive-Moving Average) neural controller requires the least computation and it's simply a rearrangement of the neural network plant model, which is trained off-line, in batch form. The only online computation is a forward pass through the neural network controller. The drawback of this method is that the plant must either be in companion form, or be capable of approximationby a companion form model (called NARMA-]L2) [10,11]. In this work, the NARMA –L2 architecture is applied with the aid of the Neural Network Toolbox of MATLAB software version-7. The identification can be summarized by the flowing steps:

a-The first step in using feedback linearization (or NARMA-L2 control) is to identify the system to be controlled. Neural network is trained to represent the forward dynamics of the system. One standard model that has been used to represent general discrete-time nonlinear systems is the NARMA-L2 model [12]:

$$y(k+d) = N[y(k), y(k-1), \ldots, y(k-n+1), u(k), u(k-1), \ldots, u(k-n+1)] \quad \ldots..(6)$$

Where $u(k)$ is the system input, and $y(k)$ is the system output and k ,d, n are integral number and N is the function of the output system after identification.

b- The next step is to make the output system follows some reference trajectory by developing a nonlinear controller of the form:

$$y(k+d) = yr(k+d) \quad \ldots\ldots (7)$$

$$u(k) = G[y(k), y(k-1), \ldots, y(k-n+1), yr(k+d), u(k-1), \ldots, u(k-m+1)] \quad \ldots\ldots. (8)$$

The problem with using this controller is : Training neural network to minimize mean square error, needs to use dynamic back propagation which quite slow [13] .

One solution is to use approximate models to represent the system. The controller used in this section is based on the NARMA-L2 approximate model:

$$\hat{y}(k+d)=f[y(k), y(k-1), \ldots, y(k-n+1), u(k-1), \ldots, u(k-m+1)] + g[y(k), y(k-1), \ldots, y(k-n+1), u(k-1), \ldots, u(k-m+1)]u(k) \quad \ldots\ldots. (9)$$

Using this equation directly can cause realization problems, because it must determine the control input based on the output at the same time, i.e: input make the system output follows the reference equation (7) . The resulting controller is:

$$u(k)=\frac{y_r(k+d)-f\,[y(k),y(k-1),\ldots,y(k-n+1),u(k-1),\ldots,u(k-n+1)]}{g\,[y(k),\ldots,y(k-n+1),u(k-1),\ldots,u(k-n+1)]} \quad ..(10)$$

Where the next controller input is not contained inside the nonlinearity. The advantage of this form is that controlled

$y(k+d)=f[y(k),y(k-1),..,y(k-n+1),u(k),u(k-1),\ldots,u(k-n+1)]+g[y(k),\ldots,y(k-n+1),u(k),\ldots,u(k-n+1)]u(k+1)$ ...... (11)

The neural network architecture is illustrated in fig. (7) [13], the testing data, validation data and training data can be shown in figures (8), (9) and (10). Figure (11) is referred to block diagram of the permanent magnet dc motor with NARMA-L2 controller , the operation performance for different speed responses with NNC is shown in figure (12). As the simulation runs, the system output and the reference signal are displayed. It's seen that the settling time of the PMDC motor is exactly reach to the value of 1 p.u. at 0.06 seconds that mean the speed of this system is very faster than without and with PI controller.

5. Conclusions

In this paper the conclusions of this work is summarized in three steps as following:

1- In the PMDC motor without controller, the motor speed is not exactly equal to the desired valuesand the speed of the motor is slowly.

2- The performance of the PI controller gives:

- Rise time: 0.09 sec. (good).
- Settling time: 0.2 sec. (good).
- Maximum overshot: 20% (poor).
- Steady state error: 0% in different speed responses (very good).

3- Neural Network Controller proposes an excellent performance as follow:

- Rise time: 0.06 sec. (very good).
- settling time: 0.06 sec. (very good).
- Maximum overshot: 0% (very good).
- Steady state error: 0% in variable speed responses (very good).

From which one can deduced that, the artificial intelligent controller, type NARMA-L2 has the best response than others because it's highly improved the speed step response. Therefore, it can be used successfully instead of complex controller methods or traditional controllers.

References

[1] David Cowden,"***Elementary Theory of DC Permanent Magnet Motors***", 10/15/2004, available via Internet from the web site at http://www.srl.gatech.edu/education/ME3110/primer/motors.htm,.

[2] Dr. C. A. Rajan, Member,(Ms) P. Geethanjali, Non–member,"***Identification and Control of PMBLDC Motor Using Neural Network***",Volume 88, December 2007, IE(I) Journal–EL.

[3] Jianxin Tang, " ***Real-Time Adaptive PID controller Using the TMS 329C31 DSK*** " Dept .of Electrical and Computer

Engineering / University of Massachusetts Dartmouth 2000.

[4] M. J. Willis, "***proportional – integral- derivative control*** ",Dept. of Chemical and Process Engineering ,university of Newcastle , November 1998 ,http:// www.lorien. Ncl.ac.uk /ming /pid.

[5] S. Weerasooriya and M. A. El– sharkawi., ***"Laboratory Implementation of Neural Network Trajectory Controller for a dc Motor"***, IEEE Transactions on Energy onversion, vol 8, no 1, 1993, pp 107–113.

[6] D. C. Hanselman, "***Brushless Permanent-Magnet Motor Design***", Mc Graw Hill, New York 1994.

[7] P. Vas, "***Parameter estimation, condition monitoring diagnosis of electrical machines***", Oxford Science Publication, *Oxford*, 1993.

[8] Cai Zixin and Xu Guanyou, "***Artificial intelligent and application***", Beijing: Qing Hua University Press. 2003.

[9] M. T. Hagan and H. B. Demuth, "***Neural Networks for Control*** ", Proceedings of the 1999 American Control Conference, San Diego, CA, pp. 1642-1656. 2009.

[10] K. S. Narenda and S. Mukhopadhyay ,"***Adaptive Control Using Neural Networks and Approximate Models***", IEEE transactions on Neural Networks, Vol.8 , 1997.

[11] K. S. Narenda and K. Parthasarathy, ***''Identification and Control of Dynamical Systems Using Neural Networks''***, IEEE Transactions on Neural Networks, Vol. 1, No. 1 , March 1990.

[12] Howard Demuth and Mark Beale, "***Neural Network Toolbox***", Copyright 1992-2008 by the Math Work, Inc.

[13] M. A. Natic, ***''Neural Networks Toolbox for use with SIMULINK''***, user's guide, The MathWorks Inc., 2008.

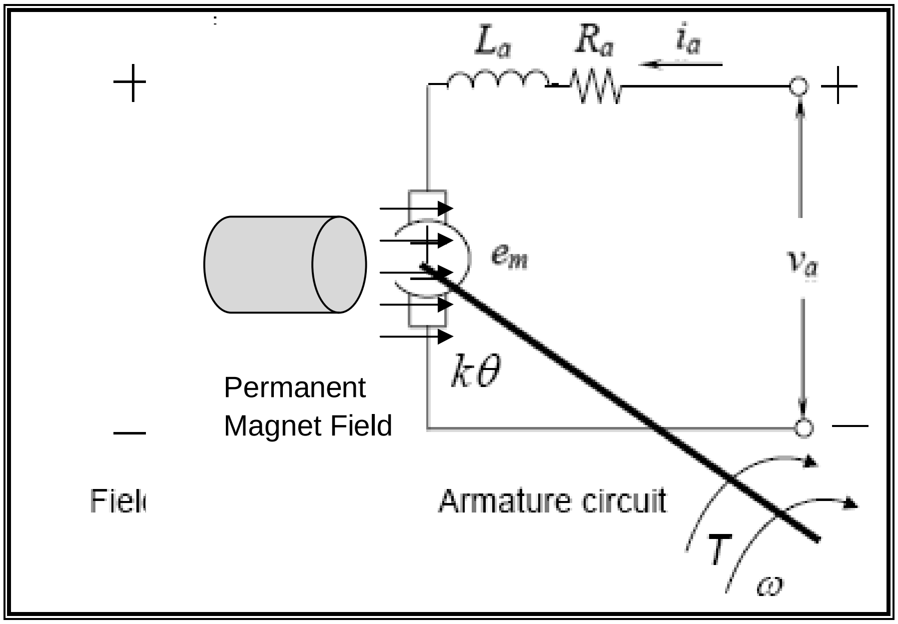

Figure (1) PMDC motor equivalent circuit.

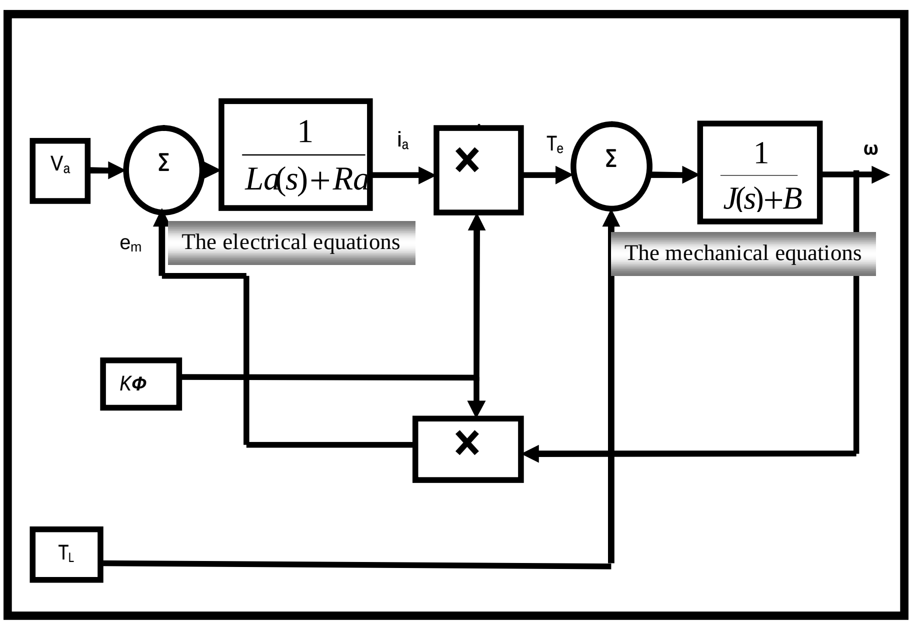

Figure (2) The block diagram of the permanent magnet dc motor

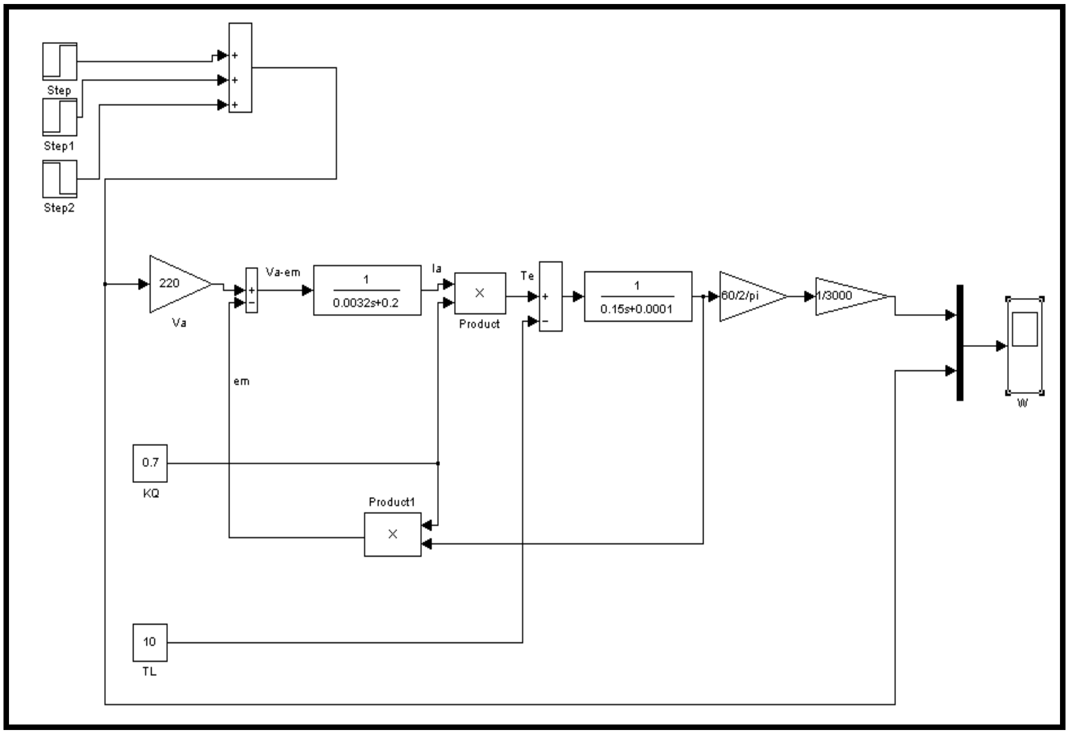

Figure (3) The transfer function block diagram of the permanent magnet dc motor without controller

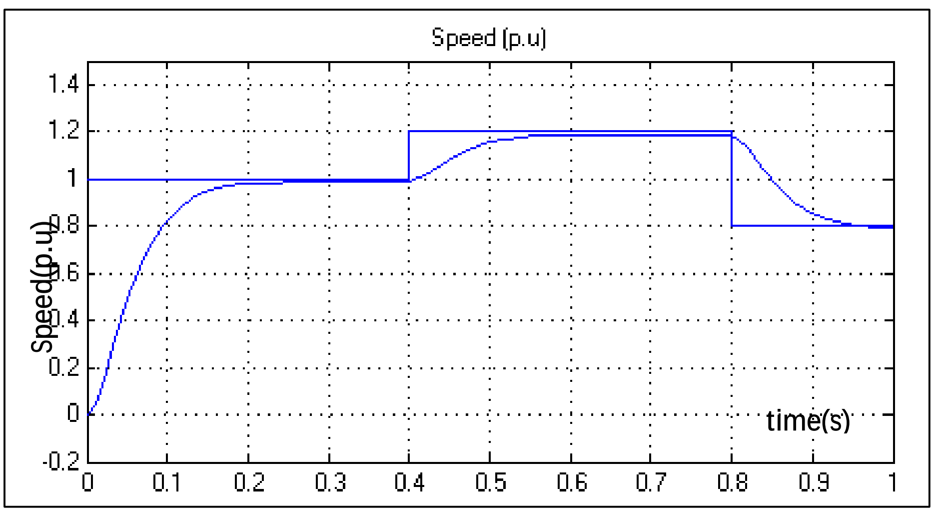

Figure (4) The operation performance for different speed responses of the permanent magnet dc motor without controller

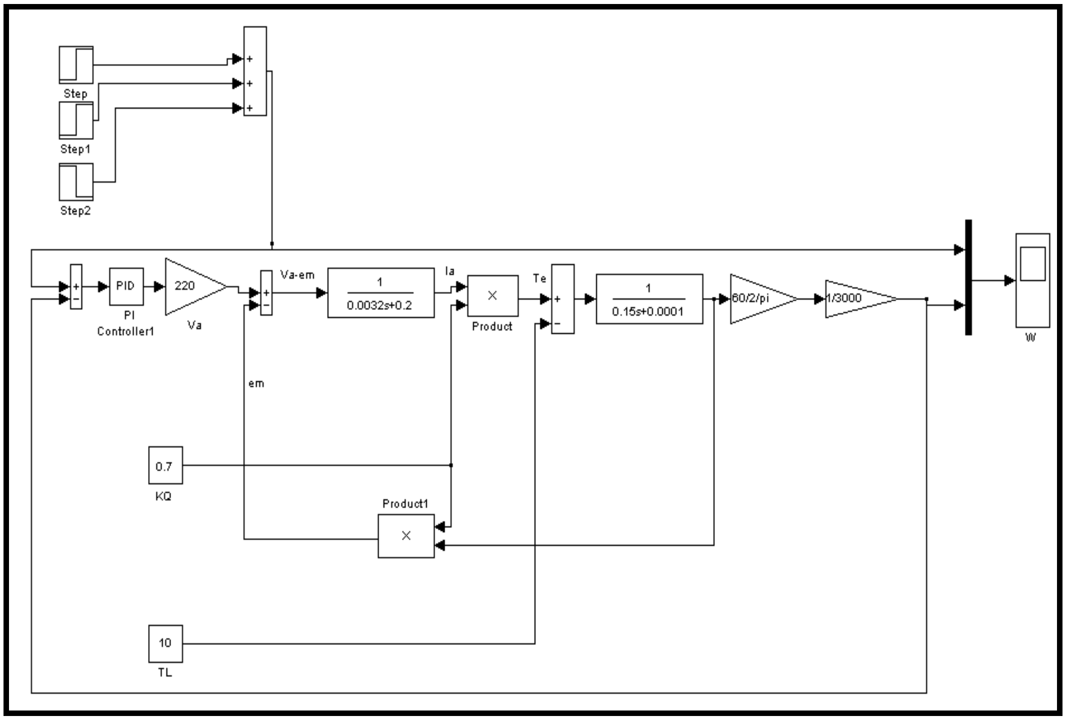

Figure (5) The block diagram of the permanent Magnet dc motor with PI controller

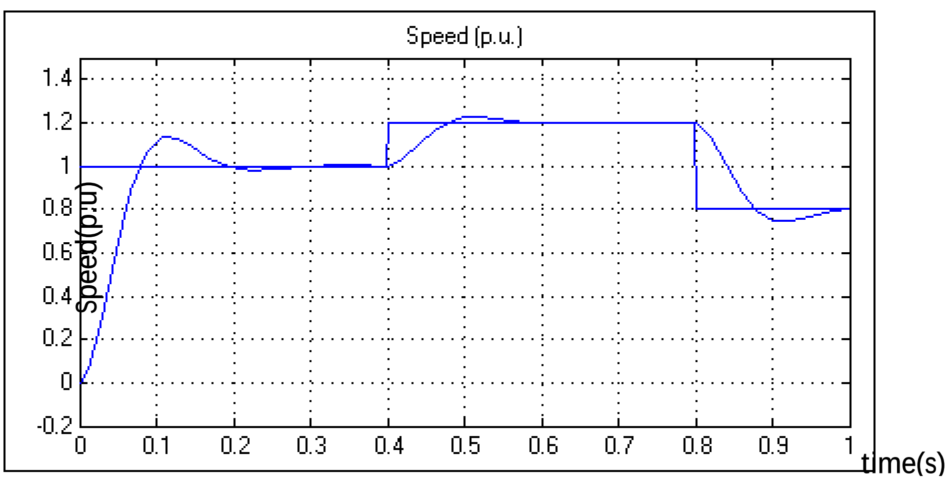

Figure (6) PI control Performance

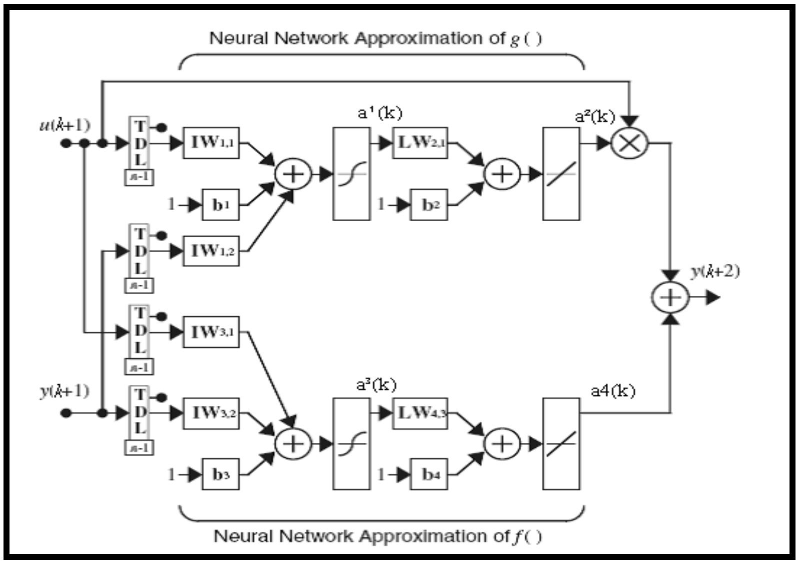

Figure (7) NARMA-L2 architecture

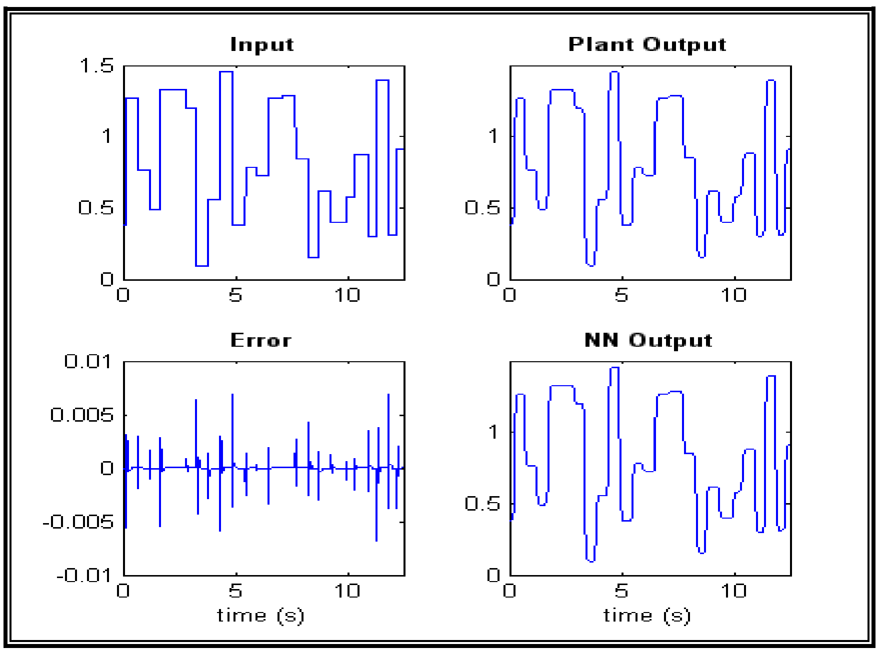

Figure (8) Testing datafor Neural network

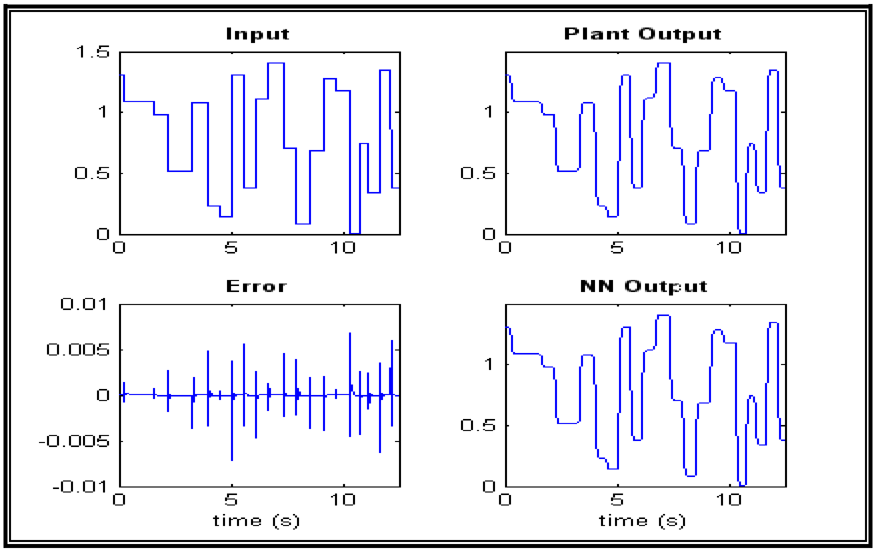

Figure (9) Validation datafor Neural network

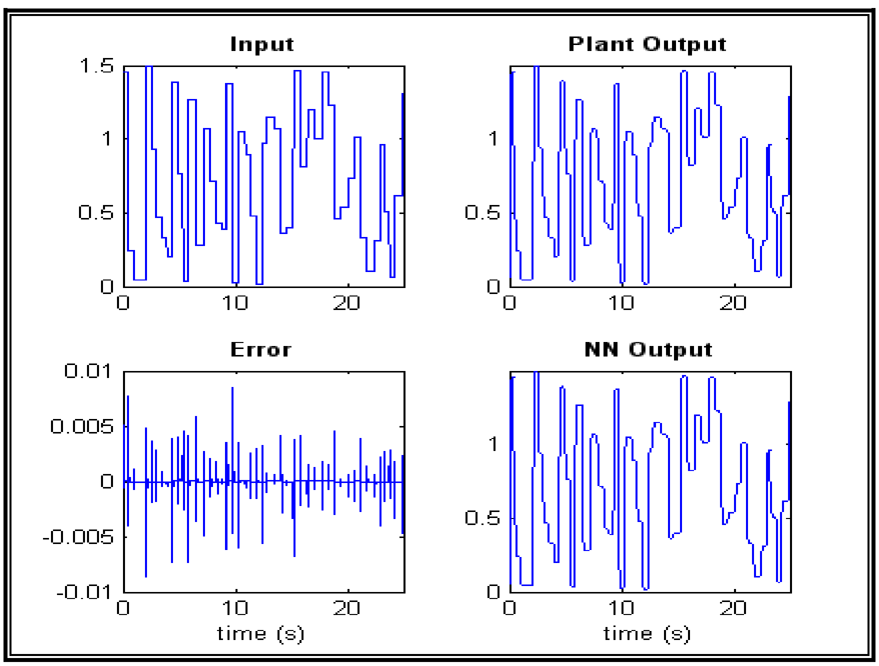

Figure (10) Training data for neural network

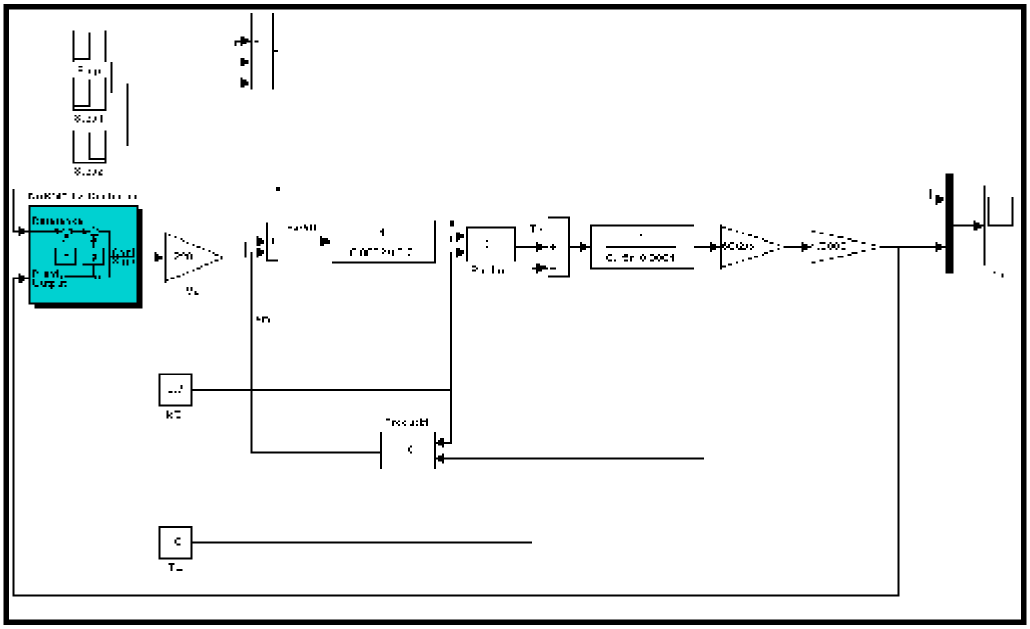

Figure (11) The block diagram of the permanent magnet dc motor With NN controller

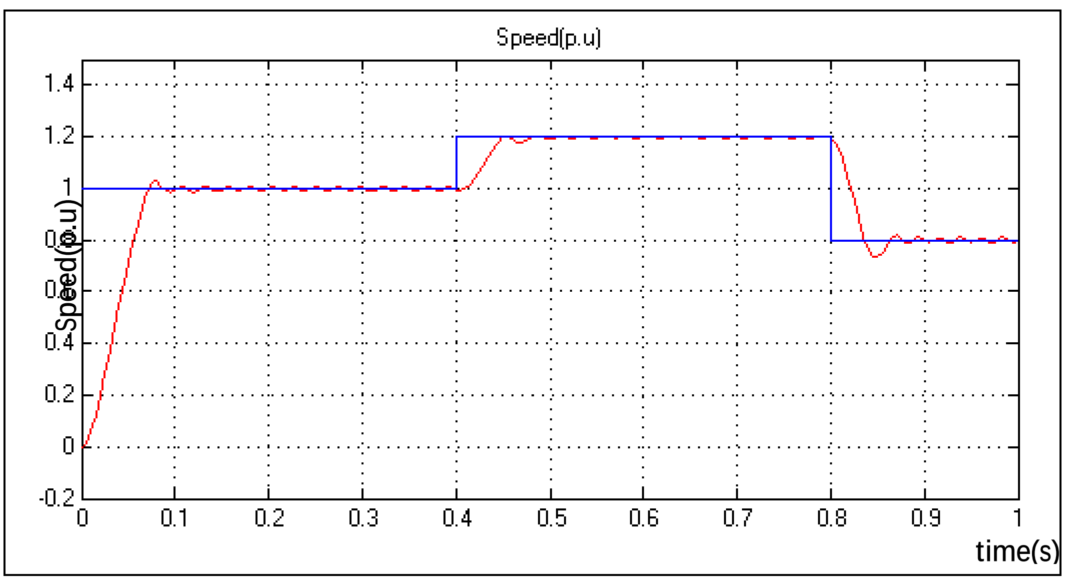

Figure (12) NN control Performance