ANN Modified Design Model to Adjust Field Current of D.C. Motor

Alia Jasim Mohammed*, Suad Khairi Mohammed* & Ahlam Luaibi Shuraiji**

Received on: 21/10/2009
Accepted on: 7/1/2010

Abstract

This work is concerned with designing an adjusted field current of D.C. motors to obtain constant speed, based on ANN. The design is employed by using training model with supervised manner with back-propagation algorithm. *MATLAB* neural network tool box is used for training purpose.

The feed-forward neural network (FFNN) and learning capabilities offers a promising way to solve the problem of system non-linearity, parameter variations on unexpected load excisions associated with D.C. motor drive system.

The proposed *ANN* controller model is implemented with a control dc motor drive system in the laboratory. The laboratory test results validate the efficacy of the based controller model for a high performance dc motor drive.

Keywords: D.C. Shunt Motor, FFNN, ANN, Controller Model, Weights and Biases, Model Training.

تصميم نموذج معدل بالشبكة العصبية الاصطناعية لتعديل تيار الأثارة لمحرك تيار مستمر

الخلاصة

يتعلق العمل في هذا البحث بتصميم نموذج يعتمد على تعديل تيار الأثارة لمحركات التيار المستمر للحصول على سرعة ثابتة، العمل مستند على الشبكة العصبية الاصطناعية (ANN) . استخدم هذا التصميم المقترح مع استعمال خوارزمية التوليد الخلفية لتدريب النموذج المقترح.صندوق أدوات (MATLAB) بالنسبة للشبكة العصبية أستعمل لأغراض التدريب. قابليات التعلم للشبكة العصبية ذات التغذية الامامية (FFNN) تعرض الحلول للمشاكل اللاخطية للمنظومة، اختلافات البارامترات على الانقطاعات الغير متوقعة للأحمال المرتبطة بمنظومة محرك تيار مستمر(D.C.).

أن نموذج جهاز السيطرة المقترح طبق على منظومة مختبرية لمحرك تيار مستمر(D.C.). صدقت النتائج المختبرية كفاءة جهاز السيطرة المستند على الاداء العالي لمحرك تيار مستمر (D.C.).

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

https://doi.org/10.30684/etj.28.11.4
2412-0758/University of Technology-Iraq, Baghdad, Iraq

1-1ntroduction

The non–linearity and the parameter uncertainty are some of the most challenging problems in the controller design of the automatic control system. For highly non-linear, bulky and complex permanent magnet excited dc motors, it may not be sometimes possible to determine the system characteristics such as non-linearity, saturation, time varying parameters, etc… . Conventional designs of robust control are often associated with constant gain controllers such as proportional integral, which stabilize a class of linear system over a small range of system parameter [1].

The adaptive controllers such as model reference adaptive controller, on the other hand adjust the control characteristics to stabilize the system precisely with unknown parameters. But implementation of the complete system in real time with these types of controllers become sometime difficult and costly, because they involve complex hardware and software computations with large number of unknown parameters [2].

The speed controller of this type of motor drive systems needs to be designed in such away so that it can handle the problems of unknown load dynamics, parameter variations and model uncertainties. On the other hand, the controller design should be as simple as possible so that it can be readily implemented in real time with minimum complexity and cost [3].

Multi-layer $FFNN_s$ (Feed Forward Neural Network) have proved extremely successful in pattern recognition, processing and speech recognition. Recently, these types of networks with back propagation training algorithm are receiving wide attention in control applications [4].

An *ANN* (Artificial Neural Network) tries to resemble very closely the biological brain structure through mathematical models by acquiring knowledge through learning. When used as a motor controller in real time, artificial Neural Network can tune itself through on-line training and instruct the drive systems same as a human operator to behave according to the desired way. This on–line training is generally based on the previous knowledge *ANN* has gathered through off-line learning [5].

Exploiting the inherent non-linear input –output mapping property of *ANN*, some controllers are being designed for dc motor drive systems with an aim to achieve the characteristics of adaptive controller [5].

This paper introduces the proposed *ANN* based adaptive controller through on-line weight up dating.

2- Model of D.C. Shunt Motor

Fig. (1) shows the conventional circuit of *D.C* (Direct Current) motor drives which consist of two parts, armature and the main field poles. The circuits of these two parts are connected in parallel and the supplied. Resistances are connected in series with both circuits to provide for necessary speed control. Excluding the effect of armature-reaction and resistance drop, the main two variables in speed control are the supply voltage and the flux [6, 7].

The basic motor equation is:-

$N = Vt - Ia\ Ra / K_f \varphi$ .... (1)

$N = Ea / K_f \varphi$,

$Ea = K_f I_f N$,

$Ea = Vt - IaRa$

Where: -

N is the speed in (r.p.m.),

Vt is the terminal voltage, at no load Ea approximate equal Vt,

Ia Ra armature resistance drop,

K_f is constant, and φ is the flux,.

It is clear that the speed varies with varies Vt, Ra and[8].

Starting from the rated speed(at no load)taken (1800 r.p.m.)by varying the resistance which is connected in series with shunt field winding shown in *fig(1)* . The motor is loaded by means of the coupled *D.C.* generator and readings of speed *(N)*, armature current *(Ia)* and torque *(T)* are noted. The torque readings are taken at each load balancing as shown in ***table (2).*** Throughout the test field current of the motor is kept constant at the value (I_{fo}) needed to run the machine on no-load-rated speed *(N)*, (I_L) , *(Vt)* and *(Vg)* (voltage generator which is equal to load voltage *(VL))* are noted. But notice that *(Vt)* for the motor and *(Vg) or (VL)* will be constant at 220 volts.

3- Feed-Forward Neural Network Structure

A general structure of *FFNN* is shown in *Fig. (2)*. An *FFNN* (Feed Forward Neural Network) consists of an input layer, an output layer and a number of hidden layer (s) between the input and output. In each hidden and output layers, there are some information processing units known as the neurons [9].

The source nodes in the input layer of the network supply respective elements of the input vectors which constitute the input signals of the second layer of the network. The output of this layer is served as the inputs to third layer of the network and so on.

The set of the output signals of the neurons in the output or final layer of the network constitutes the overall response of the network [9].

Fig. (3) Shows a model of a neuron. There are three basic elements of the neuron model, namely a set of connecting links, an odder for summing the input signal and activation function [5].

In general, the mathematical equations which are acting on a neuron *K* can be described as: - [10].

$$SK = \sum_{J=1}^{N} Wkj\ Ij \quad ....(2)$$

$$Ok = f\,(Sk + Bk) \quad ....\ (3)$$

Where $I_1, I_2, \ldots\ldots, I_N$ are the input signals; $W_{k1}, W_{k2}, \ldots\ldots, W_{kN}$ are the synaptic weights of neurons *K; Sk* is the linear combined output ; *Bk* is the bias ; *f ( )* is the activation or transfer function ; and *OK* is the output signal of the neuron. The activation function generall used is the hard limiter, piece – wise linear and sigmoid functions. In this paper log sigmoid and tan sigmoid [10].

Transfer functions used can be defined as:

$$f\,(v) = logsig\,(v) = \frac{1}{1 + e^{mv}} \quad .....\ (4)$$

And

$$f(v) = tansig(v) = \frac{1 - e^{-2v}}{1 + e^{-2v}} \quad \text{….. (5)}$$

4-Proposed Controller Model

The proposed model for our design contains two parts, the first part is the controller model and the second part is the model of *D.C.*(Direct Current) shunt motor.

4-1: Adjustment of Field Current (If)

The model training which is shown in *fig.(4)* contains the proposed *ANN* (Artificial Neural Network) controller model shown in *fig.(5)*. *T*he objective of this proposed model is to obtain constant field current by using *ANN* which uses the back propagation algorithm. The inputs and outputs of *ANN* controller can be obtained from the knowledge of conventional *D.C.* motor drive from ***table (2).***

The input patterns of our model are the constant field current*(If)* and the torque *(T)*, and the output desired for this model will adjust one value of the field from the values of the input field current shown in ***table(3).***This value of the field current which is obtained by training and as shown in ***fig. (6a).*** After that this value is used as an input for *D.C.* shunt motor with initial value of the torque *(T)*.

In this design a typical feed forward neural network structure with one hidden layer having eleven neurons has been used. This *ANN* (Artificial Neural Network) structure does not use any feedback loop which might be of concern from stability point view. The input of the training model will be the values of the field current *(If)* and the torque *(T),* these values will be taken from ***table (2)*** of the conventional *D.C.*(Direct Current) shunt motor. The output of the model training will be the field current *(If)* which is taken also from ***table (2).*** The numbers of hidden layers are chosen by trial and error, keeping in mind that the smaller the numbers are, the better it is in terms of both memory and time requirement to implement the *ANN*(Artificial Neural Network)

4-2: The conventional D.C. shunt motor

This part is contain the *D.C.*(Direct Current) shunt motor, which has the specification shown in ***table (1).*** The input of this motor will be the constant field current *(If)* which is obtained from the first part,[by training the field current (If) at(0.7 ampere) which gave the approximate the constant speed at (1800r.p.m.), training result shown in *table (3)*], and the initial value of the torque which will after that be increasing while the load is increasing. And the output of this motor will be approximate constant speed with respect to the input constant field current; by this work we obtained a *D.C.* shunt motor drive to control speed that shown in ***Fig.(6c).***

The experimental results of training shown in ***Fig. (6a).*** The simulation for weights and biases is shown in ***table (4).***

After training by using the feed forward back propagation algorithm current will be obtained constant field current. This value of the field current which is obtained after training, surely has the value of speed. This value of *(If)* will be used as the input to the *D.C.* shunt motor shown in Fig***(5)***.

5- Results and Discussions

The work is performed to obtain an initial set of weights and biases before using the *ANN* (Artificial Neural Network) in real time. Data for training can be obtained by simulation experiments for getting the data for the training purpose.

Once the input data and desired response data are available, the training can be carried out using the back-propagation algorithm. Available soft-ware can be used for training. In this paper it is generated by simulation. For adjusting the weights and biases a program has been written. The weights and biases of the model training for speed control of the motor are then adjusted according to the training algorithm described. Using the updated weights and biases the *ANN* determines the proper control field current. ***Fig. (6a)*** shows the experimental result of the training for the model training, which shows how the training reaches the goal with respect to *MSE* (Mean Square Error). From this training we get the constant field current with varied torque as shown in ***Fig. (6b).***

The constant value of the field current *(If)* obtained from the model training is taken as the input for the motor with the initial value for the torque at this point the motor works, and when the motor loaded, the speed will be constant because of the constant input value of the field current. ***Fig. (6c)*** shows the constant speed for example *(1800 r.p.m.)* with respect to variable torque.

Table (3) shows the simulated results or the output desired after training and how we reach to the final results with minimum errors. From the experiment result we notice that, for example if we want to obtain *(1800 r.p.m.),* the field current of this value of speed will be *(0.7Amp.),* so that for the model training the output desired will be *(0.7)*. And after training the value of desired will be as in ***table (3).***

The updated weights and biases for our simulation shown in ***table (4).***

Conclusions

[**] Actual experimentation can be expensive time consuming. But controlling by using *(ANN)* offers a fast and inexpensive means to learn more about speed control of a *D.C.* shunt motor. Adjustment of field current and hence the flux and speed by adjustment of the shunt field circuit resistance is accomplished simply, inexpensively, and without much change in motor losses.

[**] In this work the modeling of the *D.C.* shunt motor gave an inside look of the expected output when testing the actual *D.C.* motor. The results from the model training were approximating similar to occur in real life.

[**] Modifying the field current *(If)* and returned to the speed value approximate *(1800 r.p.m.)* at load by varying the field current rheostat, can be achieved by using *(ANN)*, this to observe the effect of the speed response of the *D.C.* shunt motor response by *Artificial Neural Network.*

[**] A Feed-Forward Neural Network is applied successfully for solving problems by training them in supervised manner with back-propagation algorithm. Basically, the back-propagation algorithm consists of two passes through the different layers of the network; one is the forward pass and the other one is known as backward pass.

References

[1] B.M. Wilamowski, "Neural Network Architectures and Learning" IEEE, Auburn university, USA, 2003.

[2] G. Lendaris "Supervised Learning in ANN_S" from "Introduction to Artificial intelligence", New York, April, 2004.

[3] S.W. Piche, "Steepest descent algorithm for Neural Network controllers and Filters", IEEE Trans. On Neural Network, Vol.5, No.2, 1994.

[4] T. Fukuda and T. Shibata, "Theory and Applications of Neural Network for industrial Control Systems", IEEE Trans. On Industrial Electronic, Vol.39, No.6, 1992.

[5] H.N. Derrick and B. Widrow, "Neural Networks for Self-Learning Control System", IEEE Control System Magazine, Vol. 10, No. 3, 1990.

[6] M.Y. Chow and Y. Tipsuwan, "Gain adaptation of networked DC motor controllers based on QOS variations", IEEE Trans Ind. Electron 50, 2003.

[7] S.J. Chapman, "Electric Machinery Fundamentals", 3rd Edition, WCB/Mc Graw-Hill, New York, 1998.

[8] Theodore Wildi, "Electrical Machines, Drives, and Power Systems", 4th Edition, Prentice Hall International, Inc., 2000.

[9] J. M. Zurada, "Introduction to Artificial Neural System", Jaico Puplishion house, 1996.

[10] W. Kinnetrock, "Neural Network," Technical University Rheinland-Pfalz, 2nd revised edition 1995.

Table (1) D.C. Motor drive Specification

D.C. motor specification	
Tipo	***160L***
Model Number	***61.5.75F***
Power	***2.94KW***
Horse Power	***4***
Voltage	***220 V***
Current	***15.4 A***
Field Voltage	***220 V***
Field Current	***1.06 A***
Speed	***1500 r.p.m.***
Wd	***Compound***
D.C. Generator specification	
Tipo	***132 M***
Model Number	***61.4.75***
Power	***3KW***
Voltage	***220 V***
Current	***13.6 A***
Field Voltage	***220 V***
Field Current	***0.35 A***
Speed	***1500 r.p.m.***

Table (2) the laboratory Results

N r.p.m	T N.m	IL amp.	Ia amp.	If amp.	VL (v)	Vt (v)
1800	0	0	2	0.7	220	240
1790	1.5	1	2.8	0.72		
1785	3.2	2	3.5	0.75		
1780	5.5	3	4.2	0.755		
1770	6.8	4	5.0	0.76		
1760	8.8	5	6.2	0.78		
1740	10.2	6	7.4	0.82		
1720	12.5	7	8.5	0.875		
1700	14.0	8	9.8	0.9		
1680	16.2	9	11.0	0.95		
1650	18.5	10	12.0	1.025		

Constant

Table (3) Training Results of (If)

If the desired After training	*If the desired Before training*
0.7000	0.7000
0.7001	0.7000
0.6998	0.7000
0.6997	0.7000
0.6999	0.7000
0.7000	0.7000
0.7000	0.7000
0.7000	0.7000
0.6999	0.7000
0.6998	0.7000
0.6991	0.7000

Table (4) Simulation for weights and biases

Weights		*Biases*
0.1121	28.4728	28.8255
0.375	10.2903-	8.3872
0.0923-	27.9948	25.9822-
0.3705-	3.6545	0.3915-
0.2923-	19.7745-	21.3208
0.4530-	14.0004	7.9854
0.2975	25.1740	22.9378
0.1878-	26.4269-	22.7147
0.2566	27.8648	22.2497-
0.5613-	9.3536-	8.6755
0.3077	24.5776	18.8493-

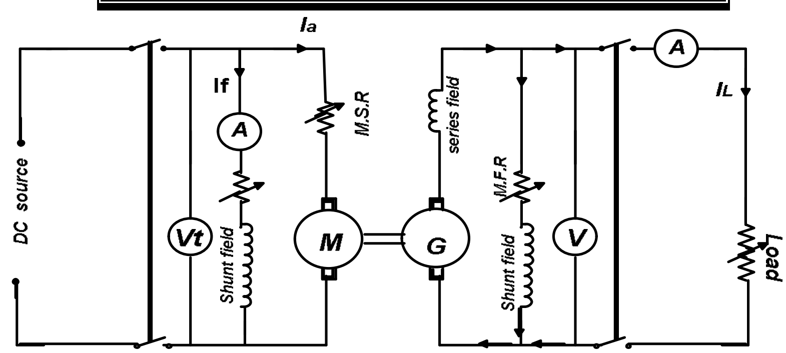

Figure (1) Conventional circuit of D.C. shunt motor [7]

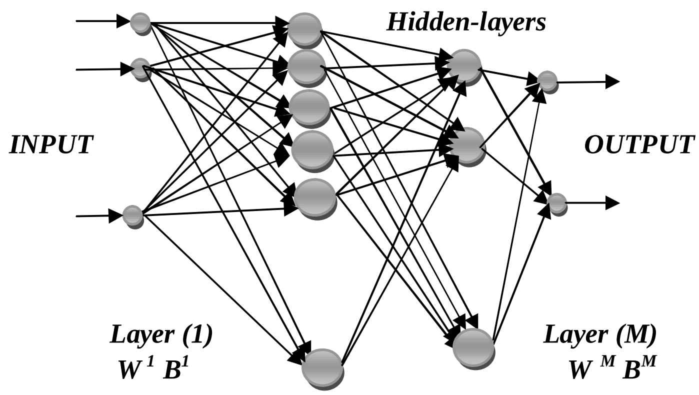

Figure (2) a general FFNN structure. [9]

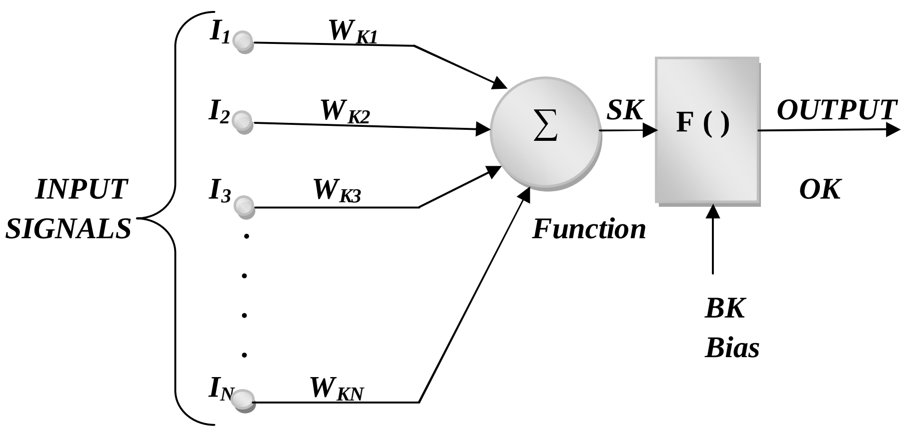

Figure (3) A neuron model. [5]

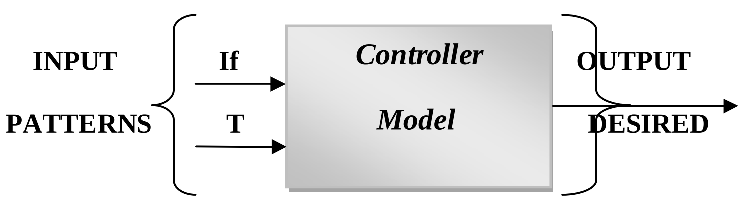

Figure (4) Proposed controller model

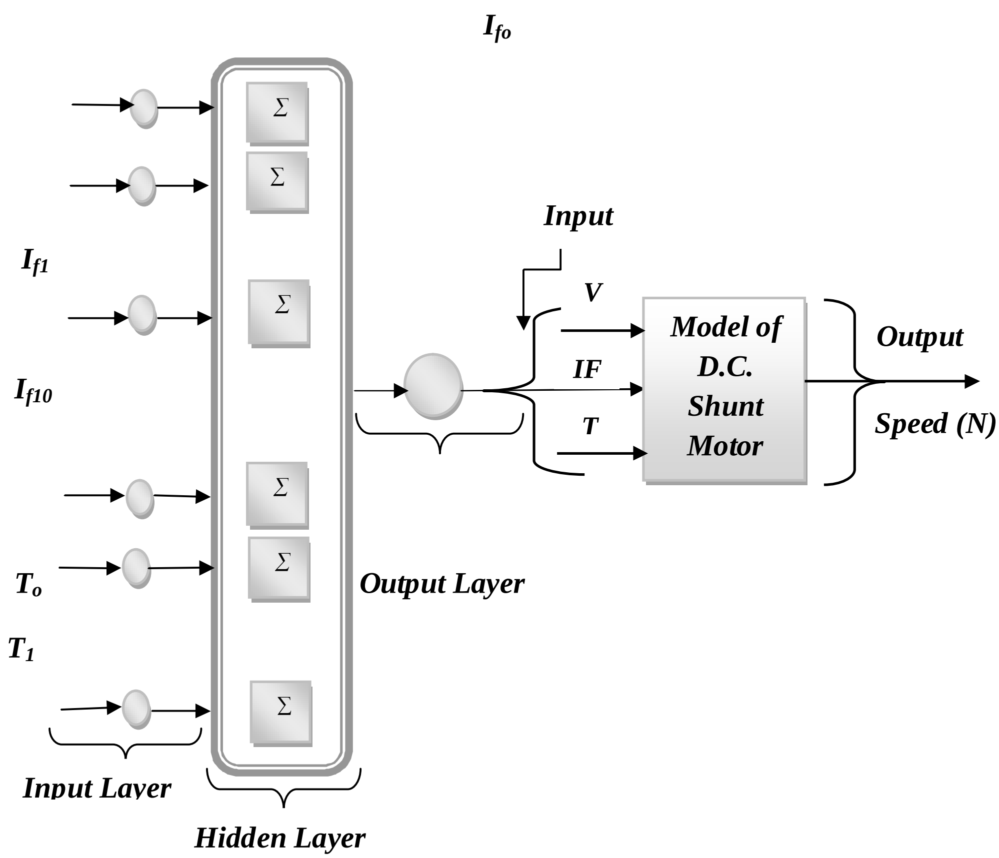

Figure (5) Proposed ANN Controller model with D.C. shunt motor

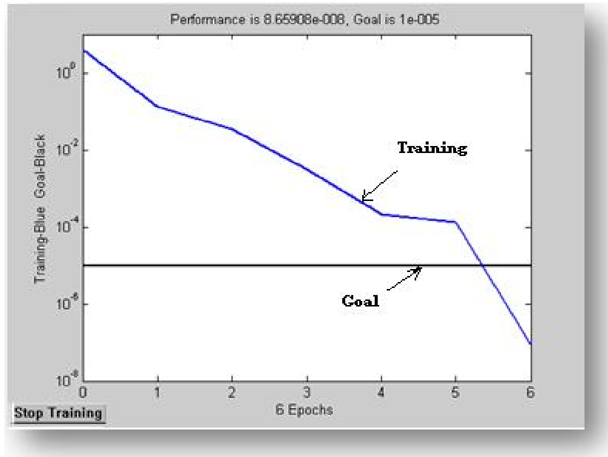

Figure (6a) Experimental results of training

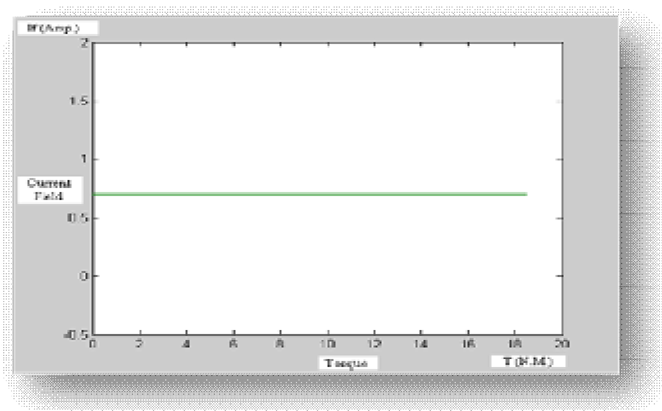

Figure (6b) Relation between current field (If) and torque *(T)*

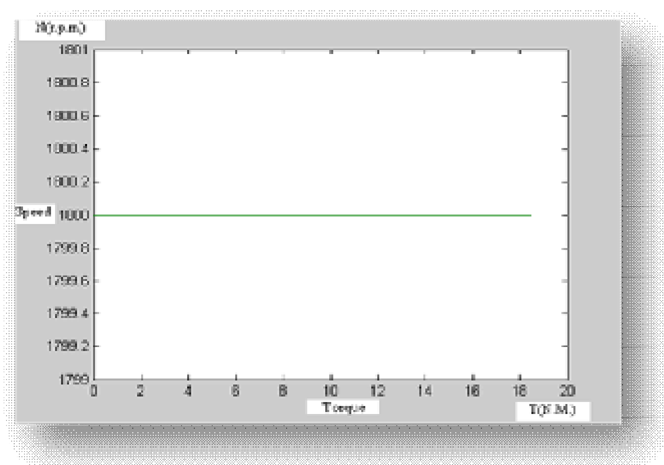

Figure (6c) Relation between speed (N) and torque (T)