P- ISSN 1991-8941 E-ISSN 2706-6703 *Journal of University of Anbar for Pure Science (JUAPS)* Open Access
2007,(1), (3) :136-139

PITTMAN MOTOR CONTROL USING NEURAL NETWORKS

WESAM M. JASIM

DEPT. OF INFORMATION SYSTEMS -COLLEGE OF COMPUTERS -UNIVESITY OF AL-ANBAR

ARTICLE INFO

Received: 12 / 4 /2007
Accepted: 3 / 9 /2007
Available online: 14/6/2012
DOI: 10.37652/juaps.2007.15599

Keywords:
Control ,
Neural Networks ,
Dynamic System..

ABSTRACT

Neural networks are well-suited for the modeling and control of complex physical systems because of their ability to handle complex input-output mapping without detailed analytical model of the systems . In this paper internal model control associated with proportional gain is used to control the system implemented with two neural networks , model of the system and inverse model.

Introduction:

In the industrial processes there are many systems having nonlinear properties . Moreover, these properties are often unknown and time varying. The commonly used PID controllers are simple to be realized , but they suffer from poor performance if there are uncertainties and nonlinearities.

The neural network controllers have emerged as a tool for difficult control problems of unknown nonlinear systems.

Since multilayer neural networks can approximate arbitrary nonlinear mapping through a learning mechanism , they can compensate the nonlinearity [1].

There are several control strategies for neural networks which some of them are as: feed forward control , direct inverse control , indirect adaptive control based on neural network identification , and internal model control (IMC) [2].

IMC requires a forward model as well as a model of the inverse of the system to be controlled , and a low-pass filter, to impact on the behavior of the closed-loop system.

The following features are special to the IMC:

- Off-set free response for systems affected by a constant disturbance.
- A requirement that the system is open-loop stable.
- It is difficult to ensure that the inverse model is trained on a realistic data set [3].

Pittman Motor:

The system to be controlled is Pittman GM9413H529 DC motor with a simulated inertial load . The simulated moment of inertia is small , and is considerably less than the actual motor moment of inertia . The equivalent circuit diagram of the DC motor system is shown in fig.(1) [4].

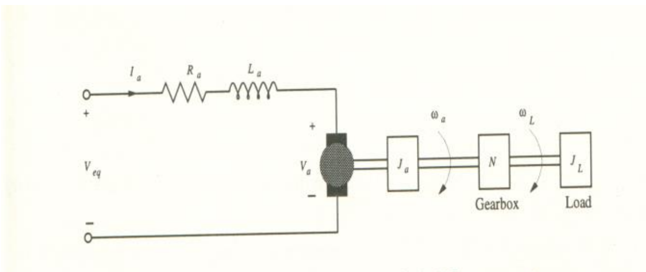

Fig.(1) DC motor circuit diagram

The transfer function of the motor can be derived from the following data:

Ra = armature resistance = 8.33Ω

La = armature inductance = 6.17 mH

Ke= back emf constant = 3.953x10-2 v/(rad/sec)

* Corresponding author at: DEPT. OF INFORMATION SYSTEMS -COLLEGE OF COMPUTERS -UNIVESITY OF AL-ANBAR, Iraq.E-mail address: wmj_r@yahoo.com

Kt = torque constant = 0.03954 N.m/A

Ja = armature inertia = 2.75x10-6 Kg.m2

JL = load inertia = 0.0137 Kg.m2

J = total inertia = 2.82x10-6 Kg.m2

N = gear ratio = 7860:18

V = input voltage = ± volt

From the above data , the following system time constants can be determined:

1/ Te = Ra / La = 1350 rad/sec

1/ Tm = Ke . Kt / Ra . J = 66.43 rad/sec Since La << Ra2 . J / Ke . Kt ,

$$G_1(s) = \frac{2.27\times10^6}{(s+1350)(s+66.4)} \quad \frac{rad/\sec}{v} \quad .(1)$$

$$G_2(s) = \frac{1}{N}G_1(S) = \frac{5194}{(s+1350)(s+66.4)} \quad ..(2)$$

$$G_3(s) = \frac{5194\times(\frac{60}{2\pi})}{(s+1350)(s+66.4)}$$

$$= \frac{49.6\times10^3}{(s+1350)(s+66.4)} \frac{rpm}{v} \quad ..(3)$$

Neural Networks For Modeling:

The use of neural networks for modeling and identification is justified by their capacity to approximate the dynamics of systems including those with high nonlinearities or dead time . In order to estimate the system dynamics , the neural network must be trained until the optimal values of the weights and biases are found . In most applications , feed forward neural networks are used , because the training algorithms are less complicated [5].

Several studies have founded that a three-layered neural networks with one hidden layer can approximate any nonlinear function to any desired accuracy [2].

The structure of three layer networks that used to identify the feed forward model and it's inverse are shown in fig.(2).[2].

It consists of an input layer , an output layer of linear activation function and one hidden layer of seven tanh units.

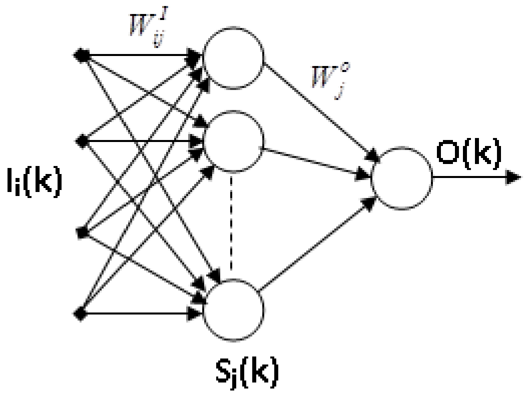

Figure (2). Neural network structure.

Where Ii(k) , Wj , Wij , Sj and O(k) are the ith input to the network , the connecting weight between jth hidden neuron and the output of the network , connecting weight between ith input to network and the jth hidden neuron , the output of jth hidden neuron and the output of the network .

This network trained using back propagation algorithm as follows :

$$S_j(k) = \sum_i W_{ij}^I I_i(k) \quad \ldots\ldots\ldots\ldots\ldots.(4)$$

The cost function

$$J = \frac{1}{2}(y_d(k) - O(k))^2 = \frac{1}{2}e^2 \quad \ldots\ldots.(5)$$

Where yd(k) and e are the desired output and the error .

$$\Delta W(k) = -\eta\frac{\partial J}{\partial W} = \eta e(k)\frac{\partial O}{\partial W}$$

$$= \eta e\frac{\partial O(k)}{\partial W} \quad \ldots\ldots\ldots\ldots(6)$$

Where W(k) is any weight of network , η is the learning rate of this weight . Therefore;

$$O(k) = \sum W_j^o S_j(k) \quad \ldots\ldots\ldots..(7)$$

$$\Delta W_j^o = \eta e(k) S_j(k) \quad \ldots\ldots\ldots..(8)$$

$$\Delta W_{ij}^I = \eta e(k)\sum W_j^o I_i(k) \quad \ldots\ldots\ldots..(9)$$

This network is trained for both the internal model and its inverse using arbitrary input data .

Control Scheme:

A control system consists of the process to be controlled and of a control device chosen by the designer , which computes the control input so as to convey the desired behavior to the control system .

The control device consists of a controller and possibly other elements (observer , filter , internal model (IM) ......) [6].

In this paper , the Internal model control system with proportional gain as shown in fig.(3) is used which characterized by a control device consisting of the controller and of a simulation model of the process .

The internal model loop computes the difference between the outputs of the process and of the IM.

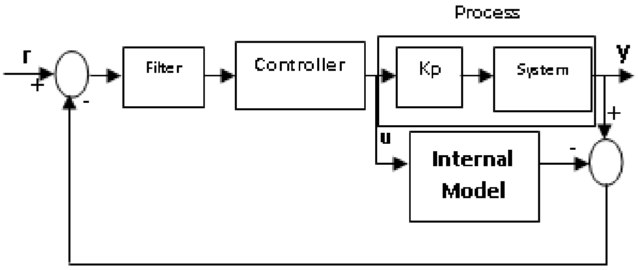

Figure .(3). The Control Scheme

This difference represents the effect of mismatch of the model . IMC devices have been shown to have good robustness properties against model mismatch in the case of a linear model of the process .

The controller (inverse model of the IM) cascaded with a low-pass filter which introduces robustness against a possible mismatch of the IM, and, though the gain of the control device without the filter is not infinite as in the continuous-time case, its interest is to smooth out rapidly changing inputs [6].

The proportional gain is added to the system to improve the output behavior of the dc motor . The convolution of the dc motor transfer function and the cascade fixed gain K_p is the overall process to be controlled .

In fig.(3) the transfer function is:

$$y(t) = \frac{FCP}{1+FC(P-M)} r(t) \qquad \text{..........(10)}$$

and

$$u(t) = \frac{FC}{1+FC(P-M)} r(t) \qquad \text{.........(11)}$$

Where;
y(t) = the control scheme output.
u(t) = the control action.
F = the low-pass filter.
C = the controller (inverse of IM).
P = the process to be controlled.
M = the internal model.

Results:

In this section simulation results are presented applying the control scheme to control the speed of the Pittman dc motor (process) .

In the first the neural network is trained to identifying the internal model and its inverse using sine wave input .

Fig.(4) shows the model response versus the process response and the inverse model versus the input to the process .

The capability of this control scheme was tested using MATLAB package by applying different input voltage level . Fig.(5) shows the input voltage versus the controlled speed of the process . In which the output of the control scheme is track the input voltage .

Conclusions:

This paper has developed a methodology to implement neural network controller for dc motor.

The neural network principles were used to construct a neural-based model for the process and its inverse.

Simulation results shows the capability of neural network controller to control the motor speed with proportional controller.

References:

[1] Mehmet Karadeniz, Ires Iskender, Selma Yuncu. "Adaptive neural network control of a dc motor". http://w3.gazi.edu.tr/web/mekara/c6-02.pdf.

[2] Hu Hongjie, Chen Jingquan, Er Lianjie. "Neural adaptive tacking control of a low speed dc servo system". Proceeding of IEEE/PES winter meeting 1991.

[3] M. Norgaard, O. Ravn , N. K. Poulsen and L. K. Hansen. "Neural Networks for modeling and control of dynamic systems". Springer-verlag, London limited, 2000.

[4] Kevin M. Passino, Stephen Yurkovich "Fuzzy control". Addison Wesley Longman, IMac. California, 1998.

[5] Mircea Lazar and Octavian Pastravanu. "A neural predictive controller for non-linear systems". www.cs.ele.tue.nl/MLazar/MATCOMpaper.pdf.

[6] Isabelle Rivals and Leon Personnaz. "Internal Model Control Using Neural Networks".

Proceedings of the IEEE International Symposium on Industrial Electronics, june, 1996, Warsaw.

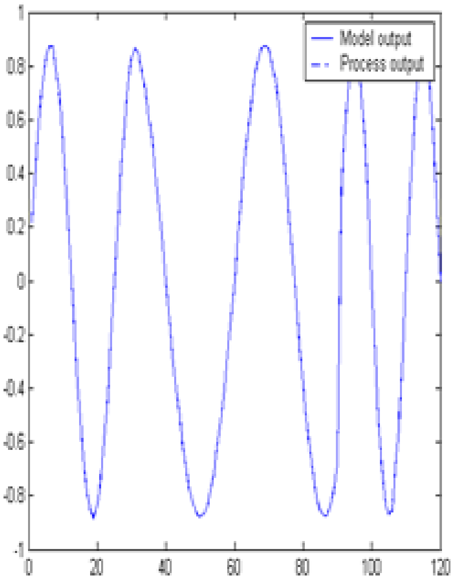

Fig.(4.a) Model output versus process output.

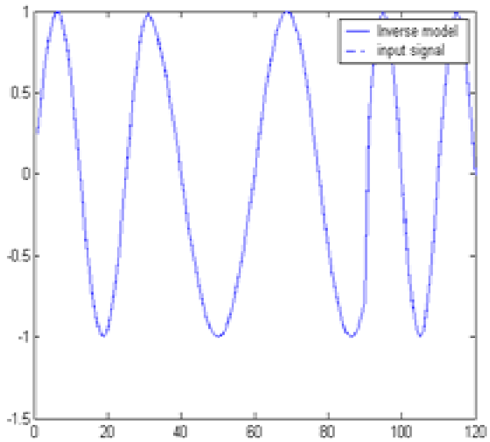

Fig.(4.b) Inverse model output versus process input.

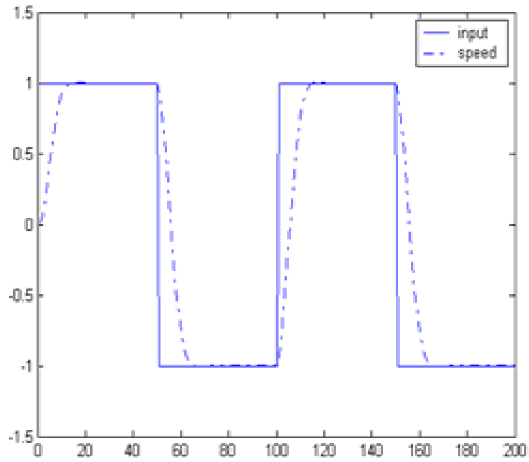

Fig.(5) The motor speed track the input.

السيطرة على المحرك المسمى (بتمان) باستخدام الشبكات العصبية

وسام محمد جاسم

كلية الحاسوب – جامعة الانبار
E-Mail wmj_r@yahoo.com

الخلاصة

تعتبر الشبكات العصبية ملائمة لتشخيص الأنظمة الفيزياوية المعقدة والسيطرة عليها بسبب مقدرتها على التعامل مع بيانات الإدخال والإخراج بدون الحاجة إلى التفاصيل التحليلية للنظام . في هذا البحث استخدم (internal model control associated with proportional gain) للسيطرة على النظام باستخدام شبكتين عصبيتين هما شبكة تمثيل النظام ومقلوبه.