Vol.5.Issue.3.2017 (July-Sept)

BULLETIN OF MATHEMATICS AND STATISTICS RESEARCH

A Peer Reviewed International Research Journal

http://www.bomsr.com

Email:editorbomsr@gmail.com

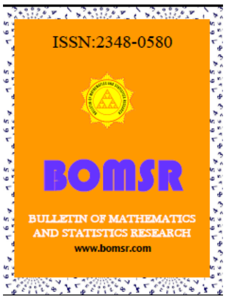
INTERNATIONAL STANDARD SERIAL NUMBER 2348-0580

RESEARCH ARTICLE

Self-tuning of PID Controllers Based on Neuro-Predictive Controller for an active magnetic bearing

Fouad Shaker Tahir Al-azawi

Lecturer, Dept of applied sciences, University of Technology, Baghdad, Iraq



ABSTRACT

This research is concerned with a new procedure of self-tuning PID controller based on neuro-perdictive control. A dreariness horizon optimal control problem is solved on line, permitting to calculate the tuning parameters of the PID controller. The proposed method is implemented on a PM electromagnets active magnetic bearing plant and a comparison with conventional auto-tuning methods is also given.

1. Introduction

Proportional-integral-derivative (PID) control is a very popular control strategy in industry due to its simple architecture and easy tuning. For many simple processes, PID control can usually obtain satisfactory control performance. However, many industrial processes possess some complex properties such as nonlinearity, and time varying properties etc.. A conventional PID controller with fixed parameters may usually derive poor control performance. .[1,2]

Model predictive control (MPC) is a name of several different control techniques. All are associated with the same idea. The prediction is based on the model of the process, as it is shown in Figure 1

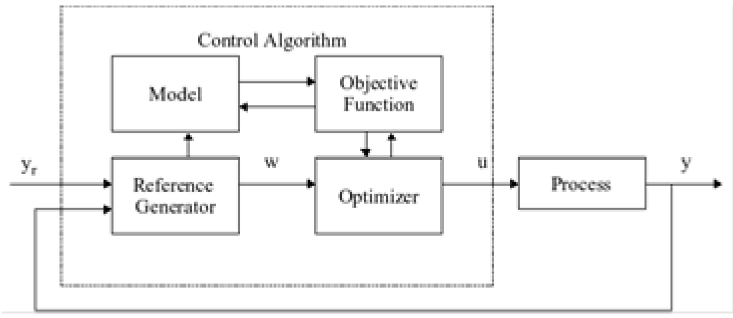


Fig. 1. Classical model-based predictive control scheme

The target of the model predictive control is to predict the future behavior of the process over a certain horizon using the dynamic model and obtaining the control actions to minimize a certain criterion. [3,4].

Neural networks have been applied very successfully in the identification and control of dynamic systems. The universal approximation capabilities of the multilayer perceptron make it a popular choice for modeling of nonlinear systems and for implementing of nonlinear controllers. The use of a neural network for process modeling is shown in Figure 2. The unknown function may correspond to a controlled system, and the neural network is the identified plant model. Two-layer networks, with sigmoid transfer functions in the hidden layer and linear transfer functions in the output layer, are universal approximators.[5,6]

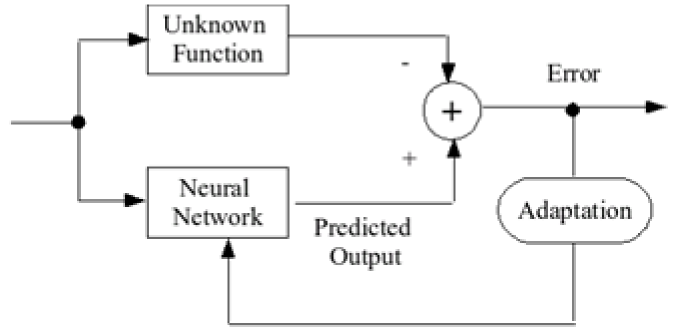


Fig. 2. Neural network as a function approximator

The prediction error between the plant output and the neural network output is used as the neural network training signal. The neural network plant model uses previous inputs and previous plant outputs to predict future values of the plant output.

2. The self-tuning procedure

The proposed self-tuning approach is based on two parallel control structures as shown in Fig. 3 that are synchronized with the reference clock of the predictable dynamics process in closed-loop with a PID controller. The upper structure uses a predictive control loop consisting of a neural predictor and a PID controller with adaptive tuning parameters. The predictive structure, with the sampling rate *Tp*, works faster than the real-time control loop supplying the predicted control error over a finite future time horizon. The tuning parameters are calculated at each sample time instant through the minimization of the predicted control error and the obtained values are used to update the tuning parameters of the real-time control loop. Thus, the controller parameters are adapted based on the Predictive optimization of the control system behavior and the desired performances can be achieved over the entire operating range.[7,8]

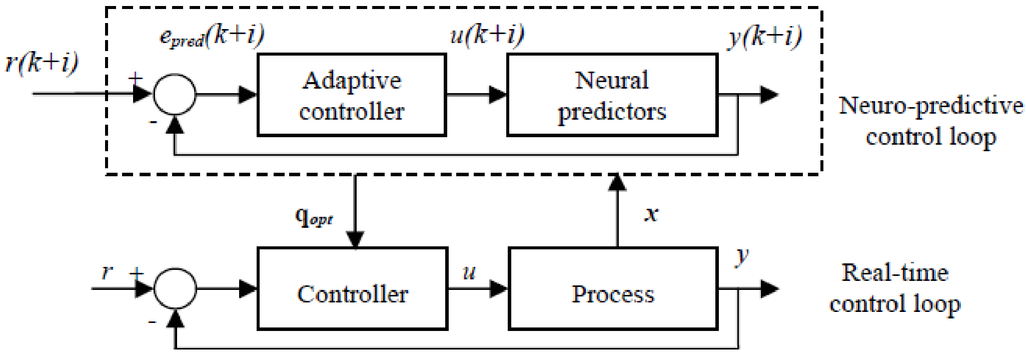


Fig. 3. Neuro-predictive structure

2.1 The design based neural network controller

In order to obtain the predictable dynamics variations at the time instants *k*, a neural predictor based on the neural-based model of the process was used. A sequential algorithm based on the knowledge of current values of *u* and *y* together with the neural network system model gives the *i*-step ahead neural predictor

$$y(k)=\sum_{j=1}^{n}W_j\sigma_j(W_j^u u(k-d+i-1)+W_j^y(k+i-1)+b_j)+b \quad \text{............... (1)}$$

The future control **u**(*k*-*d*+*i*-1*)* from (1) is obtained running the neuro-predictive control loop. Thus, at time instant *k*, the predicted output *y(k+i)* is determined, for $i = N_1, N_2$ where N_1 and N_2 are the prediction horizons. If T_p is the sampling time with which the predictive control loop operates, this must satisfy: $(N_2 - N_1)T_p << T$ .Placing the neural model of the process to operate in the neuro-predictive control loop allows for transferring the current state ***x*** of the process to the neural predictor Figure (3) at each time instant *k*. Thus, at each time instant *k,* the predicted behavior of the process is obtained in the vector form. [7].

$$y_{predictive}=[y(k+N_1)y(k+N_1+1).........y(k+N_2)]^T \quad \text{..............(2)}$$

The process output $\mathbf{y}_{predictive}$, predicted by the neural predictor, is used to calculate the predicted control error based on the controller set-point. Considering the discrete form of a PID controller,[6]

$$u(k) = u(k-1)+q_o e(k)+q_1 e(k-1)+q_2 e(k-2) \quad \text{...............(3)}$$

and the model (1), yields the following equation for the predicted control error

$$e_{predictive}(k+i)=\left(\sum_{j=1}^{n}W_j\sigma_j W_j^u u(k-d+i-1)+W_j^y y(k+i-1)+b_j+b\right)-r(k+i) \quad \text{............... (4)}$$

where the vector **u**(*k*-*d*+*i*-1*)* is a function of the tuning parameters vector **q**=[q_0 q_1 q_2]. Minimizing the cost function**.**

$$J=\frac{1}{2}\sum_{i=N1}^{N2}e^2_{predictive}(k+i) \quad \text{............... (5)}$$

3. MATHEMATICAL MODEL OF PM DC MOTOR :

In PM DC motor the field pole is permanent magnet and the flux produced by the field pole is constant. Therefore the field circuit was neglected for modeling and the armature circuit alone was considered for the motor modeling. The simulation of the entire set up was done using equation models of the motor. The PM DC motor has been modeled with the following modeling equations.[9,10]

$$\frac{di_o}{dt}=\frac{1}{L}\left[-Ri_o+V_o-K_b\frac{d\theta}{dt}\right] \quad \text{................ (6)}$$

$$\frac{d_\theta^2}{dt^2}=\frac{1}{J}\left[K_t i_o-B\frac{d\theta}{dt}+T_L\right] \quad \text{............... (7)}$$

Where :

J : Moment of Inertia of the motor.

B : Friction coefficient of the motor.

K_t : Torque constant of the motor.

K_b : Motor back e.m.f. constant.

T_L : Load torque applied.

i_o : Armature current.

V_o : Armature voltage applied.

R : Armature resistance and

L : : Armature inductance.

The Eq.(6) and Eq.(7) were derived from voltage and torque equation of PM DC motor respectively. By using the above two equations,

4. Neural model of the system

In this work, the Simulink Toolbox is used to simulate the NNPC and PM DC motor system. Figure 4 depicts the MATLAB Simulink model of PM DC motor system.[10] While, Figure 5 shows the connection between NNPC and PM DC motor system.[11,12]

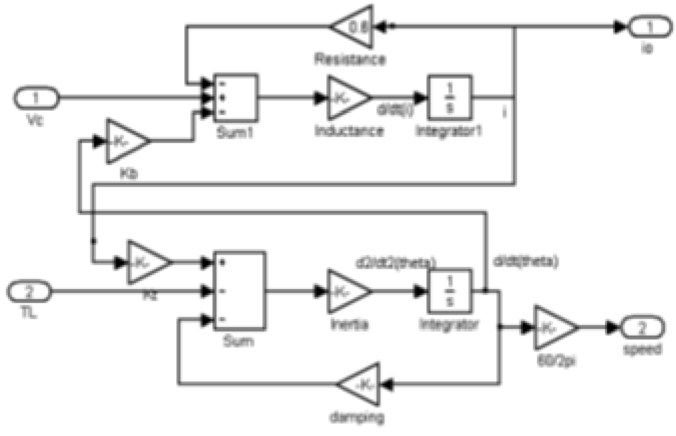


Fig.4. Simulink model for the PM DC motor

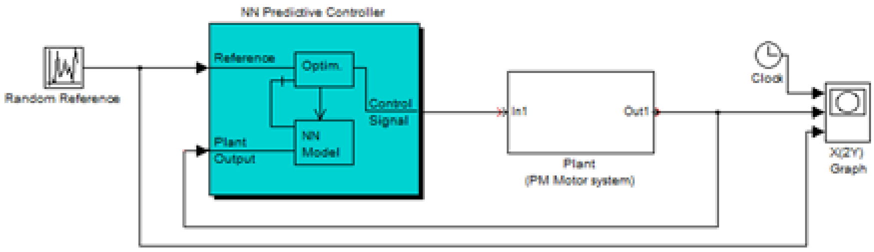


Figure. (5) ANN controller for PM DC motor

5. The Simulation Results

Neural model was created using Neural Toolbox, where we used MLP network with one hidden layer with 10 neurons and tansig activation function for modeling. We used Levemberg-Marquart method for training of the MLP network The ANN based identification architecture was implemented in MATLAB using neural network toolbox software.

Computer simulations have been carried out in order to validate the effectiveness of the proposed scheme. As mentioned previously the Neural networks are trained offline and in batch form. We have used 10 hidden layers and 8000 sample data, which are generated to train the network figure(6) show the training data generation for ANN controller in MATLAB., the data is generated to train the neural network plant controller. While data generation process, plant response follows the reference model prediction which is necessary for training's data set to be valid. IF data set is acceptable, the controller may be trained through 'Train Controller' option. The training of Artificial Neural Network controller then starts according to the given parameters.

1000 training epochs and employing training as a training function were enough to get good results. The training and validation data of the ANN controller are shown in Figs 7 and 8 respectively. A MSE performance value of 2.3689e-007 was attained for training algorithm at the maximum number of epochs (40) as shown in figure (9)

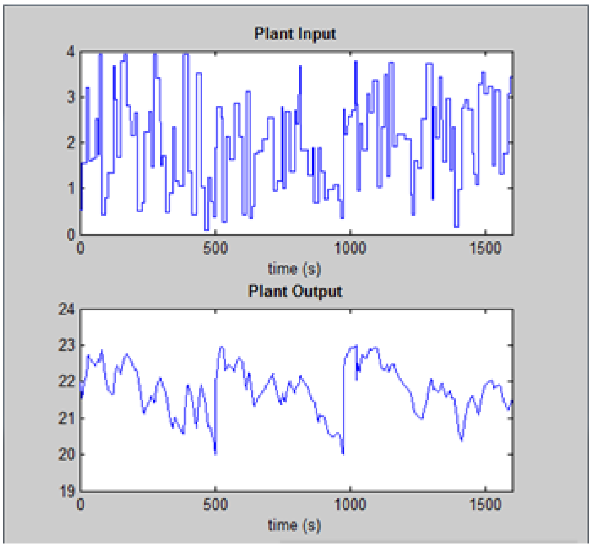
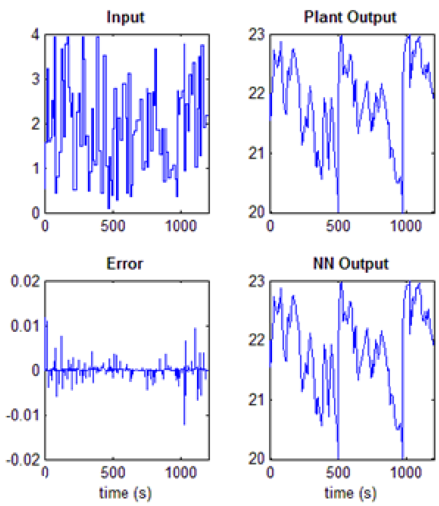


Fig. 6 Training data generation for ANN predictive controller



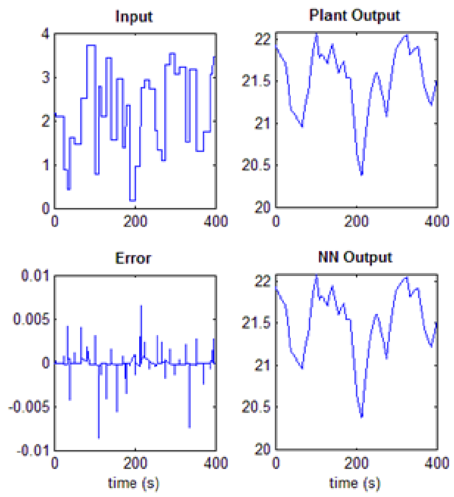


Fig. 7. training data for neural network predictive control

Fig. 8 validation data for neural network predictive control

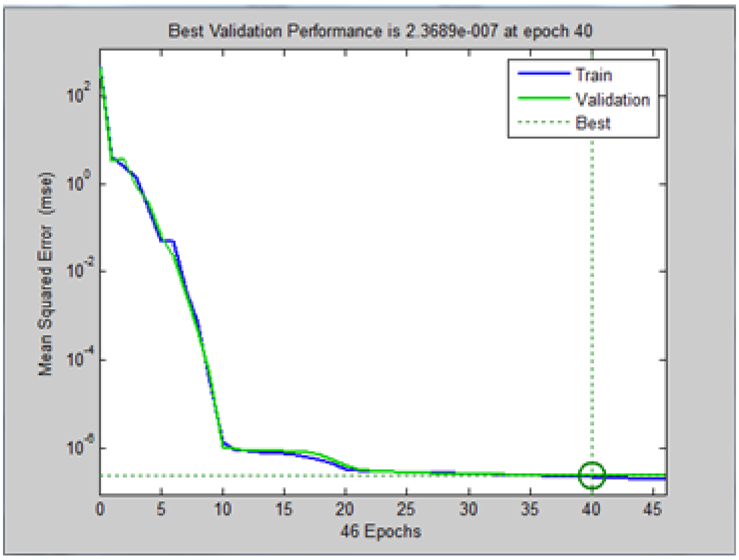


Fig. 9. Performance for neural network predictive control

6. Conclusions

A new proposed self-tuning procedure based on neuro-perdictive control has been developed. There are many advantages of this new method, like the on-line adaptation of parameters controller and its possibility to track different process operating systems, it reacts faster (with small overshoot). Using of adaptive model based on neural network is useful if we have short sampling period. This new self-tuning procedure is going to be fully implemented for real-time control application.

References

[1]. Varun A. Meng M.," A Self-Tuning Analog proportional-Integral-Derivative(PID)Controller", Automatica,Vol.44,no.1,599-609, 1999.

[2]. Dingyu X, Yang C., and Derek "P., PID Controller Design," book, www.siam.org/catalog,2007.

[3]. Shiraz Amjad, "closed loop identification for model predictive control: case study", thesis, King Fahd University of Petrloum& Minerlas, Saudi Arabia, 2003.

[4]. Ioannis Douratsos And J. Barry Gomm," Neural Network Based Model Reference Adaptive Control For Processes With Time Delay", International Journal Of information And Systems Sciences, Volume 3, Number 1, Pages 161-179, 2007

[5]. Ammar A. Aldair" Hardware Implementation of the Neural Network Predictive Controller for Coupled Tank System", American Journal of Electrical and Electronic Engineering, 2014, Vol. 2, No. 1, 40-47

[6]. Anna Vasickaninoa, Monika Bakosova"Neural Network Predictive Control of a Chemical Reactor" ,Acta Chimica Slovaca, Vol.2, No.2, 2009, 21 - 36

[7]. Varun A. Meng M., A Self-Tuning Analog proportional-Integral-Derivative(PID)Controller, Automatica,Vol.44,no.1,599-609, 1999.

[8]. corneliu lazar .sorin carari .draguna vrabic " Neuro –Predictive control based self-tuning of PID controllers" ESANN,2004

[9]. Hung C.Chen and Sheng Chang, "Genetic Algorithms Based Optimization Design of a PID Controller for an Active Magnetic Bearing", IJCSNS International Journal of Computer Science and Network Security, Vol.6 No.12, December 2006.

[10]. M. Madheswaran1 and M. Muruganandam2" simulation and implementation of PID-ANN controller for chopper fed embedded pmdc motor", ICTACT JOURNAL ON SOFT COMPUTING, APRIL 2012, VOLUME: 02, ISSUE: 03

[11]. Liuping Wang "Model Predictive Control System Design and Implementation Using MATLAB", 2009 Springer-Verlag London Limited

[12]. Manfred MorariN. Lawrence Ricker" Model Predictive Control Toolbox User's Guide For Use with MATLAB", 1998 by The MathWorks, Inc.

[13]. Corneliu Lazar, Sorin Carari, Draguna Vrabie, Marius Kloetzer, Neuro-predictive control based self-tuning of PID controllers, ESANN'2004 proceedings - European Symposium on Artificial Neural Networks Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 391-396

[14]. Yonghong Tan and Xuanju Dang Achiel Van Cauwenberghe, GENERALISED NONLINEAR PIDCONTROLLER BASED ON NEURAL NETWORKS. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.323.7309&rep=rep1&type=pdf