Artificial Neural Network based PID Controller for Control of Dynamical Systems

PAVAN KUMAR KUSHWAH
Student, Department of Electrical Engineering
Madhav Institute of Technology & Science
Gwalior (M.P.)
Pvn.kush@gmail.com

PROF. ASHIS PATRA
Professor, Department of Electrical Engineering
Madhav Institute of Technology & Science
Gwalior (M.P.)
prof_patra@rediffmail.com

Abstract—For control of various dynamical systems, an adaptive artificial neural network (ANN) based proportional integral derivative (PID) controller is developed. For linear time invariant processes, conventional PID controller is suitable but they have limitations when they are required to control the plants having high non linearity or their parameters are changing with the time. In order to find the parameters of PID controller, information regarding the dynamics of the plant is essential. Ir perturbation occurs in plant parameter(s) then PID controller may work only if these changes are not severe. But most plants are either non linear or their parameters changes with time and this demands for a use of more robust type of controller and ANN is a suitable candidate. To use the power of PID controller and ANN, ANN based PID controller is proposed in this paper. The benefit of this combination is that it utilizes the simplicity of PID controller mathematical formula and uses the ANN powerful capability to handle parameter variations and non linearity.

Keywords—Artificial Neural Network; Gradient Descent; Back Propagation Algorithm; Parameter Perturbation; Dynamical Systems; Proportional Integral Derivative Controller; Discrete Time Systems

I. INTRODUCTION

Artificial Neural Networks with the characteristics of self adapting and leaning, non linear input output mapping have been used for identification and control of both linear and non linear processes. Their effectiveness is visible when the plant is non linear and is affected by parameter variations and or disturbances. The PID controller has a simple and an elegant mathematical structure but its effectiveness is lost when system parameters changes and or system is having non linearity . In order to use the capabilities of both ANN and PID, neural based PID controller is developed. Various examples have been given in this paper where the different inputs are given at different time instants and parameter variations effects are shown to be compensated and simulation results show that the proposed controller is robust, easily realizable and is more convenient in tens of regulations of its own parameters.

II. DESIGN OF NEURAL NETWORK BASED PID CONTROLLER

There are two modes of representing PID algorithrn, one is position form (absolute form) and other is velocity form. The former gives actual value of PID controller output at the present instant while the later gives the change in the controller output at present instant. The velocity form is suitable if the final control element is a stepper motor etc. The discrete position form is shown below.

$$u_c(k) = k_p e(k) + \frac{k_p}{\tau_i} T \sum_{i=0}^{k} e(i) + k_p \tau_d \left[\frac{e(k) - e(k-1)}{T} \right] \quad (1)$$

Too many hidden layers will increase the complexity of the system and selecting few neurons in hidden layer may not provide the desired response, hence these things are decided based on the experience and the application in hand. Usually a three layered feed forward ANN (is most of the time is sufficient) consisting of single neuron in the output layer (whose output is denoted as uc (k) and only one hidden layer is used (as most cases can be handled by using single hidden layer only) with tangent hyperbolic function (since this function gives both positive and negative outputs corresponding to its positive and negative arguments unlike in sigmoid function which only give positive output) used as an activation function for the hidden layer and linear activation function (so that there is no restriction on the output range) are used for both input and output layer. The ANN weights are updated using back propagation algorithm which is based on gradient descent principle. The final weights obtained will capture the values of PID parameters. The ANN based PID controlled plant is shown in Fig .l. The ANN based PID controller is robust to parameters variations, disturbances and will adjust its weights (and hence in a way tunes PID parameters) under such scenarios.

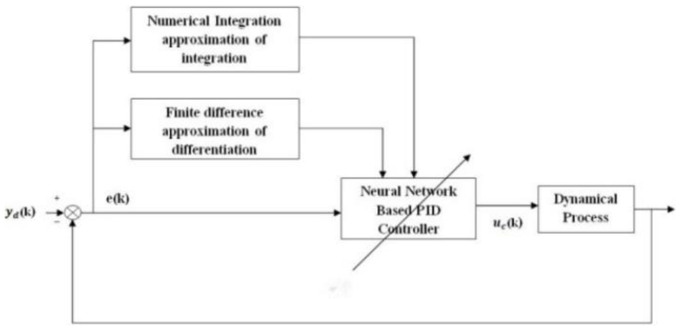

Fig. I: Structure of Process Controlled by ANN based PID Controller

III. WEIGHT ADJUSTMENT OF ANN BASED PID CONTROLLER

In order to derive the weight update (adjustment) rules for ANN based PID controller, it is important to see first the configuration of the ANN. Since the system is implemented in discrete form, hence continuous integration and derivatives in the defining differential equation of plant has to be replaced by summation and finite differences (forward difference) respectively. It is shown in Fig. 2. For deriving the weight adjustment rule we need to define first the performance index based on instantaneous error.

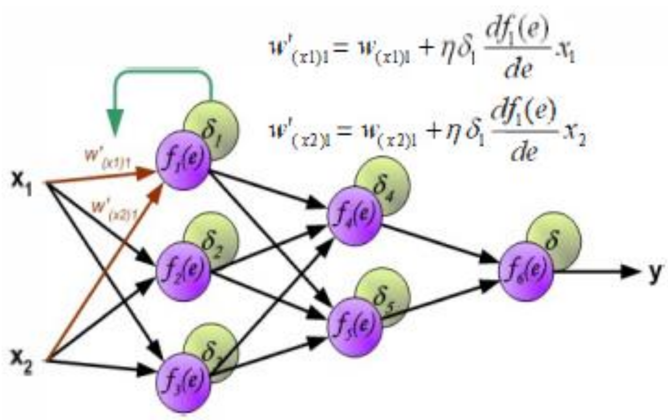

Fig. 2: ANN Structure with Update Equation

IV. STEP BY STEP DESCRIPTION OF IMPLEMENTATI ON CONTROLLER

The various steps in the chronological order are given below.

1. Decide the ANN architecture first. Application in hand will dictate the number of input and output neurons. The only thing left is to select number of hidden layers and number of neurons in them. Since most cases can be dealt by using single hidden layer so chose single hidden layer. Number of neurons must be at least twice of number of inputs.
2. Initialize the weights to small random values closer to zero and chose the value of. The 3-20-1 structure of neural network is used in this paper.
3. Sample the external desired input Yd(k) and plant output at each instant and get e(k), $T\sum_{i=0}^{k} e(i), \left[\frac{e(k)-e(k-1)}{T}\right]$ which are three inputs of ANN based PID controller.
4. Calculate the hidden layer output and final output and compute the control signal value uc (k).
5. Then using Uc (k) calculated in step 4 calculates the plant output y (k) and find the error e (k) and use it to calculate value of the performance index.
6. To compute the adjustments in the input to hidden and output layer weights respectively.
7. Repeat until termination condition is met. Usually this includes condition on MSE value to go below pre-specified small value.

V. SIMULATION RESULT :

In this section simulation results of dynamical plants using ANN Based PID controller are presented. The cases of input changes and parameter variations are also considered and comparison with conventional PID controller is also made. Each example is chosen with a different degree of complexity.

A. Second Order System

Let the transfer function model of this second order system is:-

$$G(s) = \frac{1}{s(s+1)}$$

The equivalent discrete time model of above system with sampling period of T=0.05 sec is:-

$$y(k+1) = \frac{T^2 u_c(k)}{T+1} + \frac{2y(k)}{T+1} - \frac{y(k-1)}{T+1} + \frac{Ty(k)}{T+1}$$

Clearly this is a dynamical process since its output at (k+1)th depends not only on present value of output and input but also on past values of output. The open loop step response of above system without controller is shown in Fig. (3).

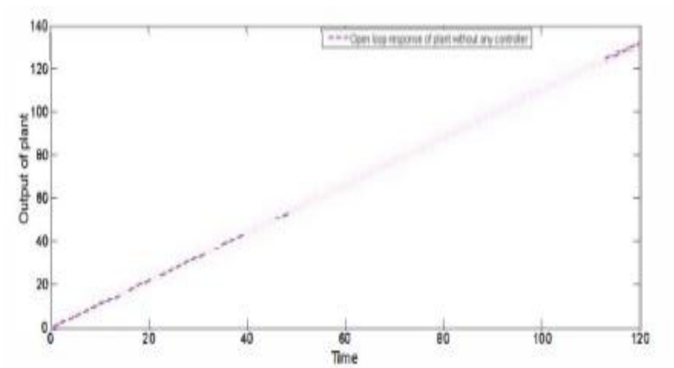

Fig. 3: Open Loop Step Response.

It is dear from the Fig. 3 that plant is not settling equal to the reference input (which is unity) value but rather its response is increasing linearly with the time. Fig. (4) Shows the comparison of plant response due to ANN based PID controller and conventional PID controller (whose parameters were tuned using the Ziegler Nichols method). For conventional PID controller the parameters values are chosen as follows: sampling period T=0.05sec and kp= 0.1, Ti= 50 and Td= 0.001. The learning rate is taken to be n= 0.06 and 20 neurons are used in hidden layer in all the examples. The controlled plant response obtained is shown in Fig. (4). It can be easily seen from Fig. (4) that response of plant when ANN based controller is connected as a controller is fast as it has smaller setting time and settles quickly to unity value (equal to reference signal) than the conventional PID controller.

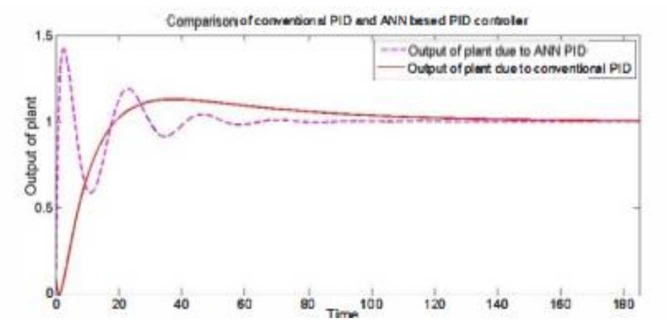

Fig. 4: Controlled Plant Response

The mean square error plot when control based on ANN based controller is shown in Fig. (5). It can be seen that as the time progresses mean square error keeps on decreasing which means weights reached to their optimum values.

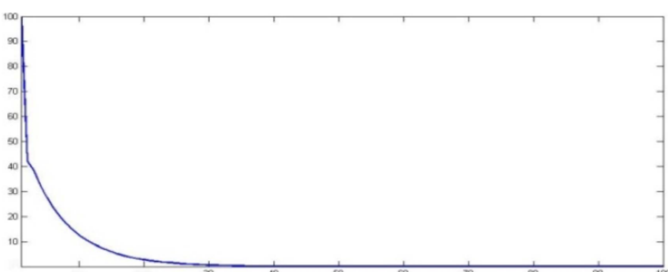

Fig. 5: Mean Square Error

Now let's check the robustness of the ANN based PID controller by doing perturbation in the value of some parameter of plant along with variation in the input's amplitude.

B. Second Order System with Single Parameter Variation and Set Point Changes

Consider the same plant with parameter T

$$G(s)=\frac{1}{\tau\, s(s+1)}$$

The parameter T initially has a value of 1.2 which is perturbed at t= 300th instant to a value 1.6 and again perturbed at t= 600th instant and t= 900th instant to values of 1.8 and 2 respectively. The sampling period of T=0.05sec is taken. At these instants the desired set point is also changed from initial value of 1 to 2, and then to 1 and -2 at t= 300^{th} t= 600th and t= 900th instant respectively. Fig. (6) Shows the online adaptive control using the ANN based PID controller (curve red) and response due to only conventional PID controller (curve black). It can be seen from the figure that variations in parameter value and changes in amplitude of input are easily handled by the ANN based PID controller. Further ANN based PID controller quickly settles and tracks the set point after parameter perturbation and changes in amplitude of input. Conventional PID controller based response took more time to settles when perturbation and set point were changed. Fig. (7) Shows the MSE plot during the online control. The spikes in the plot occurred when both input and parameter values gets changed and the resulting error quickly dies out and returned to zero since ANN adjusted its own parameters (weights) corresponding to the changes in parameter value and changes in amplitude of extern al reference input.

Comparison of plant responses due. to ANN based PIO controller and c:onventional PIO controller:-

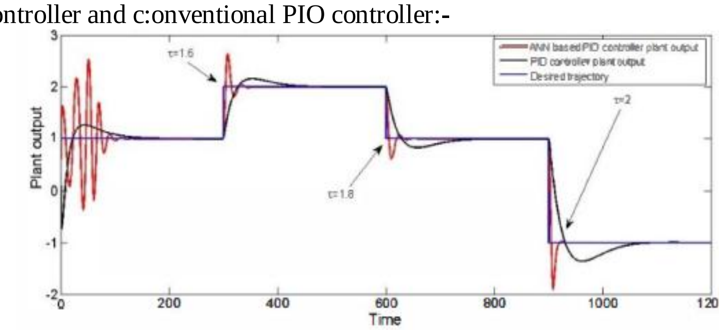

Fig. 6: Response of Plant with Controllers

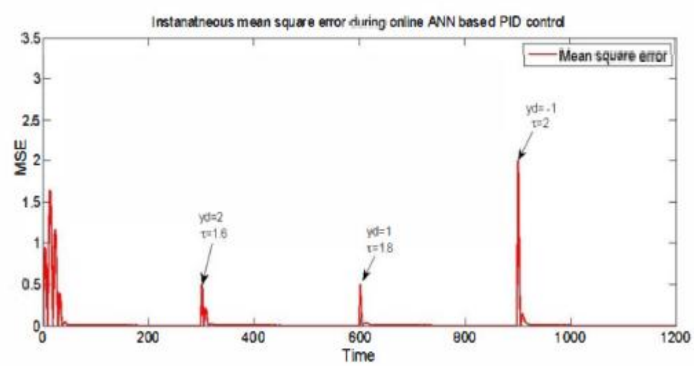

Fig. 7: Instantaneous Mean Square Error during Online Control

This happens because of the faster convergence of ANN weights to their respective desired values

C. Second Order System with Two Parameters Variations and Set Point Changes :-

Let's consider another second order system as shown below

$$G(s)=\frac{1}{\alpha s^{2}+\beta s+1}$$

Here two parameters, a, ß, of plant will be perturbed along with changes in amplitude of reference signal for testing the robustness of ANN based PID controller. The response of plant due to ANN based PID controller is compared with the response obtained with the conventional PID controller whose parameters values are kp =0:00062797, k; =0:5119and kd =0:0001 corresponding to plant's parameters initial values: a= 0 : 0001 , ß= 0 : 0002. The sampling time taken is T=0.005sec. Fig. (8) shows the comparative response plot showing that plant response (red curve) is faster when ANN based PID controller is used than when conventional PID controller is used (black curve).

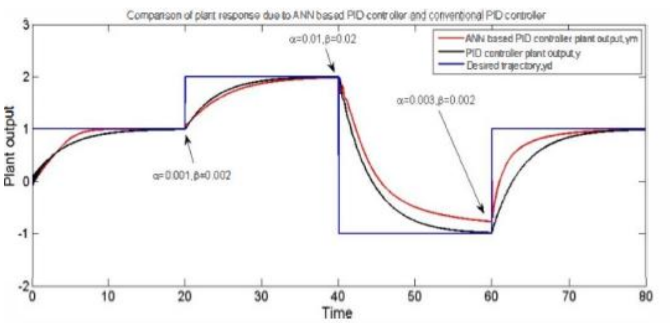

Fig. 8: Comparison of Responses of Plant

Figure (9) shows the instantaneous mean square error during the online control of plant when ANN based PID controller is used. The spikes in MSE plots at times.

T= 20th, 40th and 48th. Secs are due to parameter perturbation and reference input amplitude changes.

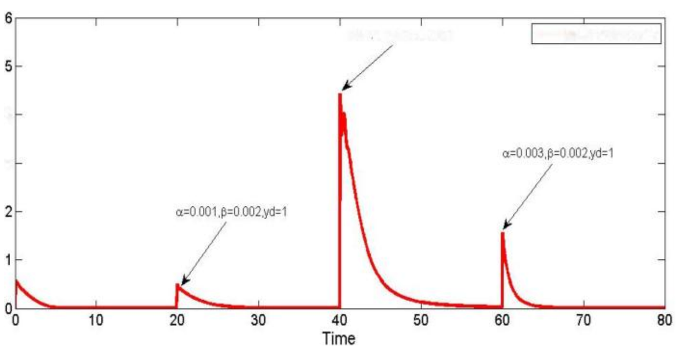

Fig. 9: Mean Square Error

VI. CONCLUSION

In this paper, ANN based conventional PID controller is implemented. The weights update equations are derived using the powerful principle of gradient descent. Further, ANN structure considered is also simple since it contains only one hidden layer along with input and output layers. Even if we don't know the sampling period of given system, ANN can approximate it along with the parameters of PID during the online control in the form of its own weights values. ANN based PID is found to be more robust than conventional PID controller to parameter perturbations and set point changes, hence it can be very useful for the control of non linear industrial processes where conventional PID won't be able to perform efficiently.

VII. REFERENCES

[1] K.S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," Neural Networks, IEEE Transactions on, vol. I, no. I, pp. 4-27,1990.

[2] Y. Yongquan and H. Ying, "Zeng bio a pid neural network controller," in Neural Networks, Proceedings of the International Joint Conference on, vol. 3, 2003.

[3] Y. Pan, P. Song, and K. Li, "PID control of miniature unmanned helicopter yaw system based on rbf neural network," in Intelligent computing and Information Science. Springer, 2011, pp. 308-313.

[4] H. Zhang, Y. Shi, and A. S. Mehr, "Robust PID control for multivariable networked control systems with disturbance/noise attenuation," International Journal of Robust and Nonlinear Control, Vol. 22, No. 2, pp. 183-204,2012.

[5] G. Stephanopoulos, "Chemical process control: an introduction to theory and practice," 1984.

[6] M. Gopal, Control systems: principles and design. Tata McGraw Hill Education, 2002.

[7] M. Singh, S. Srivastava, J. Gupta, and M. Handmandlu "Identification and control of a nonlinear system using neural networks by extracting the system dynamics," IETE journal of research, vol. 53, no. I, pp. 43-50, 2007.

[8] M. Singh, S. Srivastava, J. Gupta, and M. Hanmandlu, "A type-2 fuzzy neural model based control of a nonlinear system," in Cybernetics and Intelligent Systems, 2004 IEEE Conference on, vol. 2. IEEE, 2004, pp. 1352-1356.

[9] S. Srivastava, M. Singh, M. Hanmandlu, and A. Jha, "Modeling of non-linear systems by fwnns and their intelligent control, International journal of adaptive control and signal processing, vol. 19,no.7,pp.505-530,2005.

[10] S. Srivastava, M. Singh, M. Hanmandlu, and A. N. Jha, "New fuzzy wavelet neural networks for system identification and control," Applied Soft Computing, vol. 6, no. I, pp. 1-17,2005.

[11] M. Singh, S. Srivastava, M. Hanmandlu, and J. Gupta, "Design of interval networks based on neural network and coquet integral," Applied Soft Computing, vol. 11, no. I, pp. 138-153 , 2011.

