Simulation of Neuro-PID Controller for Pressure Process

Athi thilagalakshmi R[1]
PG Student
Karunya University
Coimbatore

Vijay Anand L D[2]
Assistant Professor
Karunya University
Coimbatore

ABSTRACT
PID controller has been designed based on neural network using Back Propagation (BP) algorithm. NNPID controller output is applied to Pressure control in a tank. A Neural Network weights are equivalent to PID controller parameters and NN adjusts the weights based on the operational status of the system in order to achieve better performance. This paper provides simulation results of Neural Network PID controller for Pressure Process using MATLAB and LabVIEW. From the simulation, NNPID controller has many advantages compared to conventional PI controller like less over shoot, more rise time and less settling time.

Keywords
Pressure control, PI controller, Neural Network, BP algorithm

1. INTRODUCTION
The PID controlling is one of the first control strategies. The PID controller is widely used in the process control and the motion control, especially in the systems for which can be established the precise mathematical model. However, the performance of it depends on *kp*, *ki*, *kd*. Over the years, many tuning methods for the PID controllers have been introduced, such as the Ziegler-Nichols tuning formula, internal model control method and so on. But, with the development of the industry, the plants are more and more complex, nonlinear complex systems. The conventional PID controller could not achieve the desired control effect because of the unsuitable parameters. Therefore, the application of conventional PID control is much more restricted and challenged [1].

When the system has external disturbances, such as the variations of output pressure, changing process dynamics, then the transient response may go down. For this reason, intelligent control schemes have proposed.

The major advantages of Back Propagation Neural Network over the traditional controller is that it can tune the PID parameters online without requiring the prior knowledge of the mathematical model of different plants [2].

In recent years, the current interest has been focussed on design of self tuning controller by Neural Networks (NN)[3]. It can be applied in two different techniques, one is to use the NN to adjust the parameters of PID controller and the other is to use it as a direct controller. PID parameter values can also be adjusted by creating NN system based on the system output error signal [2].

In the present paper we have analysis the conventional PI controller and Neural Network PID controller. In Neural Network PID controller, NN weights equivalent to PID controller parameters, are trained to achieve better control than existing conventional PID.

2. EXPERIMENTAL SETUP
The pressure process station consists of two cylindrical tanks horizontally. One of the tanks is kept at a higher level than the other. In this project as shown figure 1 in the schematic of pressure control, the pressure tank 2 is the controlled variable and the input flow rate of air to the tank 1 is manipulated variable. The tank 1 and tank 2 are connected serially. The pressure in tank 2 is measured using pressure transmitter and given to current to voltage converter, which converts the 4-20 mA current signals to 0 to 5 V signal.

The control system computes the error between the set point and the measured variable and the controller calculates the control action which is given to plant through the DAC. The output of the DAC is in the range of 0-5 V is given to the voltage to current converter which gives an output in 4-20 mA. The current signal is given to pressure converter which gives to the current to pressure converter which gives an output in 3-15 psi pressure. The pressure signal is given to the pneumatic control valve which moves the stem position to vary the input flow rate of air to the tanks.

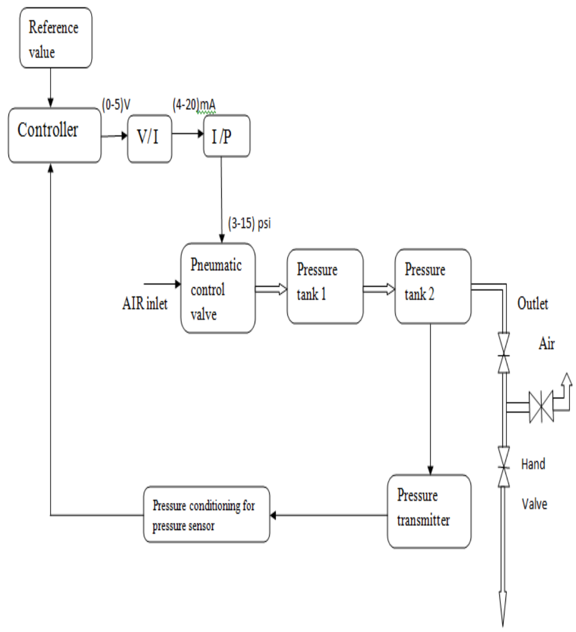

Figure 1: schematic diagram of a pressure process station

To start with compressor, it has been switched on and the air flow from the compressor was allowed continuously to the pressure tank to reach the pressure set point. The pressure transmitter measures the tank pressure and gives an output current signal (4-20 mA) which will be converted to 0-5 Volts using a current to voltage converter.

3. PID CONTROLLER

The proportional integral derivative (PID) controller is the common controller is the industrial closed loop control system. The PID algorithm can be given by,

$$U = k_p \, Perr + ki \int Perr \, dt + k_d \frac{dPerr}{dt} \qquad (1)$$

Where k_p, ki, k_d is respectively the coefficient of proportion, integral and differential [4].

4. NEURAL NETWORK PID CONTROL

4.1. Fundamental Structure of NNPID controller

Neural network could be used to regulate the parameters of the PID controller. The PID control system based on BP neural network was composed by the conventional PID controller and BP neural network, its structure is shown in figure 2, k_p, k_i, k_d are regulated by the neural network.

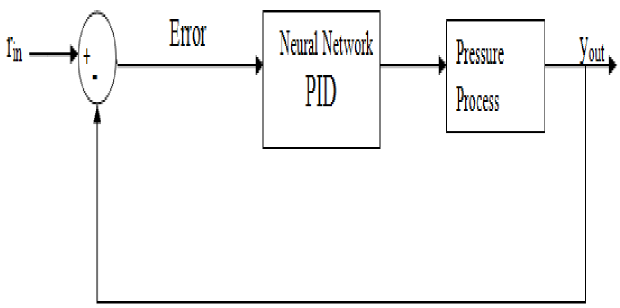

Figure 2: the structure of the PID controller based on BP neural network

4.2. Neural Network Design

The BP neural network can contain different hidden layers. But it is theoretically proved that three layered network with unlimited nodes can fit any non linear mapping. The number of input layer neurons is two namely I_1 and I_2 that is Pressure set point (P_s), Pressure output (P_o). The output of the network is PID controller output O_1. Therefore the number of neurons in the output layer is one. The hidden layer has three neurons and they represented as H_1 (P-neuron), H_2 (I-neuron) and H_3 (D-neuron).

According to the block diagram, the actuating error P_{err} can be expressed as,

$$Perr = Psp - Pop \qquad (2)$$

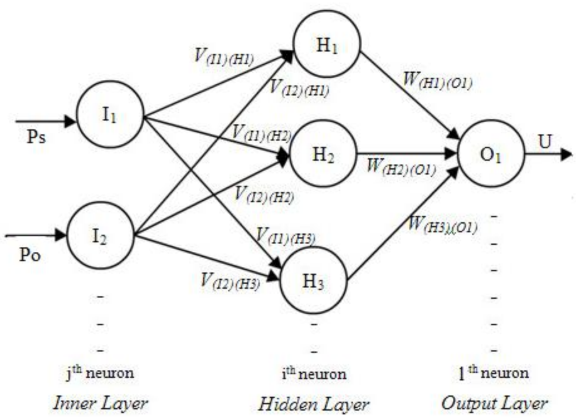

Figure 3: BP network structure

Weights between inputs to hidden layer are,

$$V_{(I1)(H1)} = +1 , V_{(I2)(H1)} = -1 \qquad (3)$$

Weights between hidden layers to output layer are taken in terms of PID parameters,

$$W_{(H1)} = K_p, \; W_{(H2)} = K_i, \; W_{(H3)} = K_d \qquad (4)$$

Input to hidden layer nodes are given as,

$$X_{(H1)} = V_{(I1)(H1)} I_1 + V_{(I2)(H1)} I_2$$

$$= Psp\text{-}Pop = Perr \qquad (5)$$

$$X_{(H2)} = V_{(I1)(H2)} I_1 + V_{(I2)(H2)} I_2$$

$$= Psp\text{-}Pop = Perr \qquad (6)$$

$$X_{(H3)} = V_{(I1)(H3)} I_1 + V_{(I2)(H3)} I_2$$

$$= Psp\text{-}Pop = Perr \qquad (7)$$

The outputs of the hidden layer nodes are equal to their inputs, which can be expressed as function of proportional, integral and derivative as given below [2]:

$$Y_{(HI)} = Perr \qquad (8)$$

$$Y_{(H2)} = \int Perr \, dt \qquad (9)$$

$$Y_{(H3)} = \frac{dPerr}{dt} \qquad (10)$$

Then, input to output layer becomes,

$$X_{(O1)} = W_{(H1)(O1)} X_{(H1)} + W_{(H2)(O1)} Y_{(H2)} + W_{(H3)(O1)} Y_{(H3)}$$

$$= k_p \, Perr + ki \int Perr \, dt + k_d \frac{dPerr}{dt} \qquad (11)$$

Where $Y_{(HI)}$, $Y_{(H2)}$ and $Y_{(H3)}$ are output coming from hidden layer nodes, $X_{(O1)}$ is input part of output layer.

4.3. BP Neural Network learning Algorithm

In the conventional BP multi-layered neural network, the learning algorithm is the gradient descent algorithm in which the gradient is calculated by back propagation method, according to the rules of windrow-Hoff. BP algorithm is used for weighting coefficients.

The aim of the algorithm is to minimize the error as given in equation (12) in order to recover the system quickly from the effects of the external disturbance by tuning of PID parameter.

BP neural network learning algorithm is described as follow:

$$E_K = 0.5*[P_{SP}-P_{OP}]^2 \tag{12}$$

The weights of hidden nodes are adjusted by BPNN algorithm based on steepest descent online training process. It is done in terms of adjusted weights of hidden layer to output layer $[W_{IL}]$ and input layer to hidden layer $[V_{IJ}]$. The increments of weight in hidden to output connection are updated by gradient descent method as given by,

$$\Delta W_{jl}(n) = -\eta \frac{\partial Ek}{\partial W_{jl}} + \alpha \Delta W_{jl}(n-1) \tag{13}$$

$$\frac{\partial E_K}{\partial W_{jl}} = \frac{\partial E_K}{\partial Y_{(ol)}} \times \frac{\partial Y_{(ol)}}{\partial X_{(ol)}} \times \frac{\partial X_{(ol)}}{\partial W_{jl}}$$

$$\Delta W_{jl}(n) = -\eta[\frac{\partial E_K}{\partial Y_{(ol)}} \times \frac{\partial Y_{(ol)}}{\partial X_{(ol)}} \times \frac{\partial X_{(ol)}}{\partial W_{jl}}] + \alpha \Delta W_{jl}(n-1) \tag{14}$$

Where, η - Learning rate, α - Inertia coefficient

$$\Delta W_{jl}(n) = -\eta[P_{SP} - Y_{(ol)}]\ Y_{(ol)}[1 - Y_{(ol)}]\ Y_{(Hi)} + \alpha \Delta W_{jl}(n-1)$$

$$\delta_l = [P_{SP} - Y_{(ol)}]\ Y_{(ol)}[1 - Y_{(ol)}] \tag{15}$$

$$\Delta W_{jl}(n) = -\eta\ \delta_l\ Y_{(Hi)} + \alpha \Delta W_{jl}(n-1) \tag{16}$$

Weights update for input to hidden layers as given as

$$\Delta V_{jl}(n) = -\eta \frac{\partial E(k)}{\partial v_{jl}} + \alpha \Delta V_{jl}(n-1) \tag{17}$$

$$\frac{\partial E_K}{\partial v_{jl}} = \frac{\partial E_K}{\partial Y_{(ol)}} \times \frac{\partial Y_{(ol)}}{\partial X_{(ol)}} \times \frac{\partial X_{(ol)}}{\partial Y_{Hi}} \times \frac{\partial Y_{(Hi)}}{\partial X_{(Hi)}} \times \frac{\partial X_{(Hi)}}{\partial v_{jl}}$$

$$\Delta V_{jl}(n) = -\eta \gamma_i \beta Y_{(Hi)}[1 - Y_{(Hi)}]\ I_{(ij)} + \alpha \Delta V_{jl}(n-1) \tag{18}$$

$$\Delta V_{jl}(n) = -\eta\ \delta_k\ I_{(ij)} + \alpha \Delta V_{jl}(n-1) \tag{19}$$

Where,

$$\gamma_i = W_{il} \delta_k$$

$$\delta_k = -\gamma_i \beta Y_{(Hi)}[1 - Y_{(Hi)}]$$

New weights for hidden to output layer is,

$$\Delta W_{jl}(n+1) = -\eta\ \delta_k\ Y_{(Hi)} + \alpha \Delta W_{jl}(n) \tag{20}$$

New weights for input to hidden layer is,

$$\Delta V_{jl}(n+1) = -\eta\ \delta_k\ I_{(ij)} + \alpha \Delta V_{jl}(n) \tag{21}$$

The new weights are changed by updated weights as per equation (20) and (21) with iterations till get the minimum mean square error in terms of pressure.

5. SIMULATION RESULTS

The algorithm and step of PID controller based on BP neural network is described as follow:

1. Choosing the structure of the BP neural network, selecting the number of node for each layer and giving the initial value of the learning speed rate and inertial coefficient;
2. Reference input (Psp) , and process output (Pop), then calculating error, Perr = Psp – Pop;
3. Calculating input and output of node cell of each layer of BP network. The weights of output layer in each node is namely three adjustable parameters of PID controller k_p, k_i, k_d;
4. Calculating output of PID controller;
5. According to BP neural network learning, online adjusting weight and achieve PID parameter and control the process of the system;
6. Setting k= k+1, return to step 1.

In simulation method using MATLAB SIMULINK module is designed for PI controller. In LabVIEW simulation, PI controller and NNPID are designed.

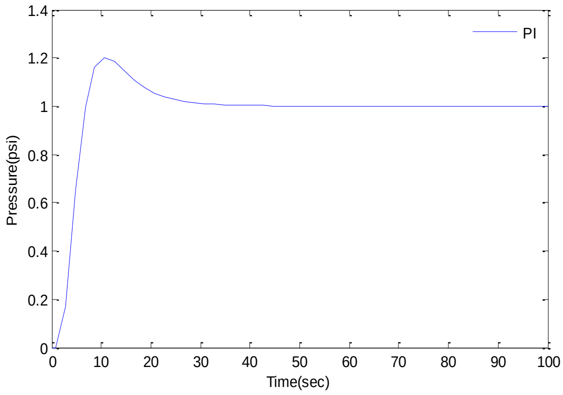

Fig 3: PI controller output in MATLAB

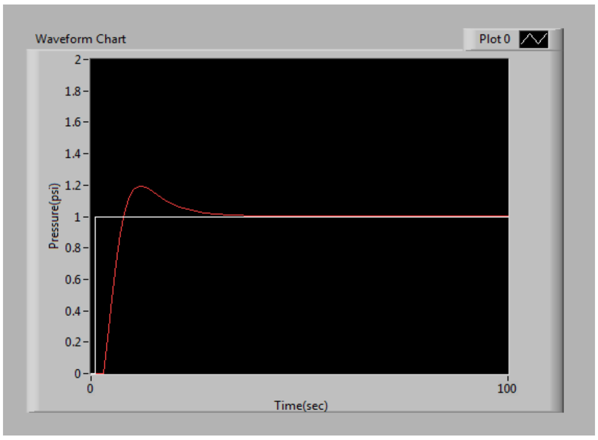

Fig 4: PI controller output in LabVIEW

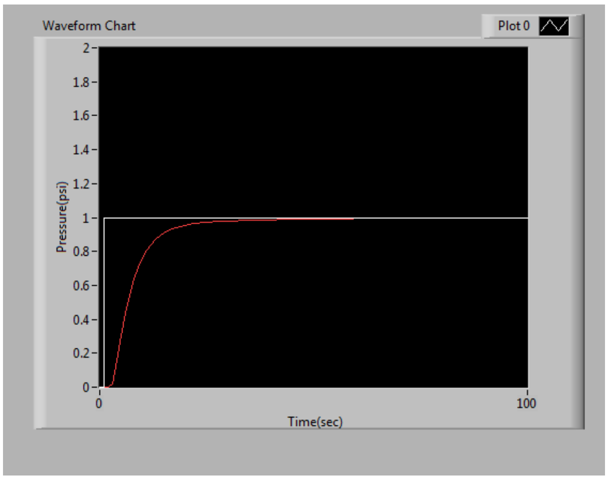

Fig 5: NNPID controller output in LabVIEW Simulation

The system output maintains the reference input and the performance of the proposed controller in figure 5 is better than that of the conventional PI controller.

6. CONCLUSION

PID controller based on BP neural network is an effective method combining traditional PID control and advanced neural network algorithm. It overcomes the disadvantage of PI control and provides a useful method for complicated system. The simulation results of pressure control in a tank by using PID controller based on BP neural network show that it can better control robustness and also reduce the settling time, over shoot and better rise time. And also PID parameter will change according to the set point. So neural network based PID controller based on BP algorithm is more adapt to the control process of pressure in a tank.

6. REFERENCES

[1] LIU Jinhao, KAN Jiangming, 2009. "Self-Tuning PID controller Based on Improved BP Neural Network", second international conference on Intelligent Computation Technology and Automation.

[2] Nisha Jha, Udaibir Singh, T.K.Saxena, Avinashi Kapoor. 2011. "Online Adaptive Control for Non Linear Processes Under Influence of External Disturbance", International Journal of Artificial Intelligence and Expert System, volume(2), issue(2).

[3] Kenji Iwasa, Noboru, Sigeru Omatu.1995."Pressure control in a plant Generating chloride by Neural network PID control", Neural Networks Proceedings., IEEE International Conference on publication.

[4] Beitao Guo, Hongyi Liu, Zhong Luo, Fei Wang.2009. "Study of PID Neural Network For Hydraulic system", Proceeding of the IEEE International Conference on Automation and Logistics.

[5] Quing-Guo,Wang, Tong-hengLee, Ho-WangFung, QiangBi and Yuzhang. 1999. "PID Tuning For Improved Performance", IEEE Transactions on Control System Technology.

[6] Zazilah May, Muhammad Hanif Amaran, Nor Aisyah Ghazali. 2011. "Neural Network Predictive Controller for Pressure Control", second international conference on Instrumentation, control and automation.

[7] Mohammed Y. Hassan, Ganesh Kothapalli. 2010. "Comparison between Neural Network Based PI and PID Controllers", seventh International Multi-Conference on Systems, Signals and Devices.

[8] Jie Hao, Wenxiang Zhang, Mingqing Ling.2010. "Research on PID Control Algorithm of BP Network based on Two Axis Digital Control Turntable", Tenth IEEE International Conference on Computer and Information Technology.

[9] F.Shahraki, M.A.Fanaei, A.R.Arjomandzadeh,2009. "Adaptive System control with PID Neural networks" ,Chemical engineering transaction.

[10] Jiangjiang Wang Chunfa Zhang, Dawei An and Youyin Jing,.2007."Study of Neural Network PID Control in Variable frequency Air-conditioning System", IEEE International Conference on Control and Automation.

[11] Hua Ji, Zhiyong Li, 2009. "Design of Neural Network PID controller Based on Brushless DC Motor", Second international conference on Intelligent Computation Technology and Automation.

[12] Liu Luoren, Luo Jinling, 2011. "Research of PID control algorithm based on Neural Network",sciverse science direct energy procedia.

[13] M.K.O.Ayomoh, M.T.Ajala,2012. "Neural Network modeling of a tuned PID controller", European journal of scientific research, volume.71.no:2.pp:283-297.