Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 54 (2015) 207 – 214

Procedia Computer Science

Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)

Multilayer Neuro PID Controller based on Back Propagation Algorithm

Rahul Patel* and Vijay Kumar

Department of Electronics and Communication Engineering, IIT Roorkee, Haridwar 247 667, India

Abstract

The paper covers the basic idea, mathematical features of conventional Neuro PID controller which is a technique to make any linear or nonlinear system unaffected to unpredictability of system's parameters and disturbances such as noise. Here we suggest a technique to apply neural networks for the tuning of the PID (proportional, integral and derivative) controller's gains in a way human tune the gains depending on the environmental and systems requirements. Error back-propagation method is used as the tuning method for the controller which is also known as BP method and this method works on the local minima algorithm.

Peer-review under responsibility of organizing committee of the Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)

Keywords: MLP (Multilayer Perceptron); Neuro controller; Neural network; PID controller; ZN (Ziegler-Nichols).

1. Introduction

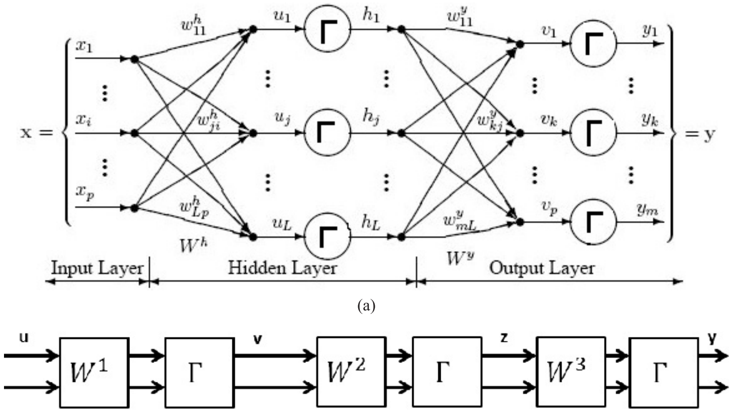
The neural approach came to existence when the typical and conventional ways to control the system failed to tackle the problems like speech, vision and pattern because real world cannot be represented in mathematical expressions.

To identify any neural network or any neural computer, it should consist three basic features: controller must have a large numbers of simple processing units and these units are called neurons in neural networks, each and every processing unit or neuron should be connected to a large number of other processing units, and the functionality of the network is determined by modifying the strengths of the connections during a learning phase. In real world, humans as well as machines can be used to perform any control function on any system but in general there are fine and keen distinctions between human and industrial Control Systems[1–5]. As an example, human employs very large amount information form sensors of the controller in planning and improving the performance of the controller in contrast of machine control system. The information processing procedure is the second difference between human and industrial controller. In the last we will discuss the most important difference, is that human control is dependent on learning process, while the working of industrial controller is governed by a predefined algorithm.

In short we can conclude a basic neural network based controller that is designed should have three basic as well as important features: the utilization of all the information from sensors of the controller, and it should have collective processing capability, and in the last it should be able to adapt the result of controller.

*Corresponding author.
E-mail address: rahul444patel@gmail.com

Peer-review under responsibility of organizing committee of the Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015)
doi:10.1016/j.procs.2015.06.023



(a)

(b)

Fig. 1. (a) Three layer neural network; (b) Block diagram of three layer neural network.

The motive behind the Multilayer Neuro PID is to stabilize an unstable stable (Magnetic Levitation which has one of the pole in right half of the s-plane) with very small settling time and overshoot as well as zero steady state error. The major motive behind Multilayer Neuro PID is to make the system response fast.

2. Multilayer and Recurrent Network

2.1 Multilayer networks

An orthodox type of multilayer neural network basically has three layers, an input layer, an output layer and two hidden layers as shown in Fig. 1. For simple accessibility block diagram form is shown in Fig. 2 with three synaptic weight matrices W_1, W_2, and W_3 and a diagonal nonlinear operator 'r' with identical sigmoidal elements

$$\alpha[\text{i.e.,}\quad \alpha(x)] = [1 - e - x]/[1 + e - x] \tag{1}$$

Neural network shown in the Fig. 1, the synaptic weight matrices and multilayers which are input, output and hidden layers are symbolized by the operator

$$N_i[u] = \Gamma[w_i u] \tag{2}$$

The mapping of input vectors and outputs of multilayer neural network in mathematical equations is shown as

$$\begin{aligned} y &= \Gamma[u] = \Gamma[W_1\Gamma[W_2\Gamma[W_1 u]]] \\ &= N_3 N_2 N_1[u] \end{aligned} \tag{3}$$

2.2 Recurrent networks

Recurrent networks suggest another approach for recognition of pattern and Hopfield advised this approach so it is called Hopfield network shown in Fig. 2 which has single layer network N, with discrete time delay in feedback that is represented as

$$x(k+1) = N_1[x(k)], X(0) = x_0 \tag{4}$$

For any system with a feedback has a diagonal transfer matrix having same diagonal elements $1/(s + \alpha)$ in continuous time system, then system in mathematical equation is as:

$$x = -\alpha x + N_1[x] + I \tag{5}$$

where I is the identity matrix which works as threshold value.

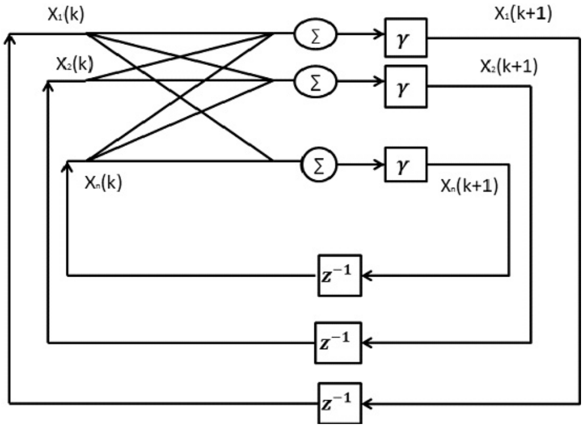


Fig. 2. Recurrent hopfield network.

2.3 Generalized neural network

There are only three elementary operations are required for the formulation of the recurrent network and that operations are:

1) Delay
2) Summation
3) The Non-linear operator N_i

The delay component in the case of continuous-time network is substituted by integrator and a constant can also be multiplied according to the need of the system.

3. PID Controller

There exists several ways to form a PID controller but the most suitable and well known PID controller is:

$$u(t) = K\left[e(t) + 1T_i \int e(t)dt + T_{dde(t)dt}\right] \tag{6}$$

where,

$$\begin{aligned} e(t) &= \text{desired output – actual output} \\ K &= \text{proportional gain} \\ T_i &= \text{integral time} \\ T_d &= \text{derivative time} \end{aligned}$$

The most accepted representation of PID controller with an input and an output is on view.

4. A General Parameter Adjustment Method: Gain Scheduling Method

The Gain Scheduling Method is basically has nonlinear type feedback which has linear parameters changed depending on the operation conditions and output from the previous conditions for the controller (Fig. 3, and Fig. 4). The way of controlling looks very simple theoretically but real time implementation is possible for only for computer controller because it's difficult to obtain previous conditions without using computer. In Fig. 5, the parameters are

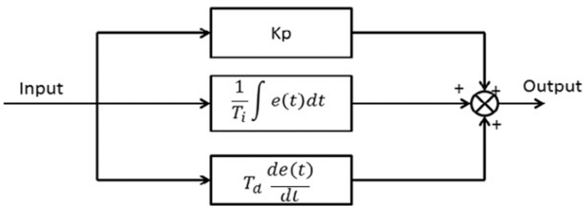


Fig. 3. Basic form of PID controller.

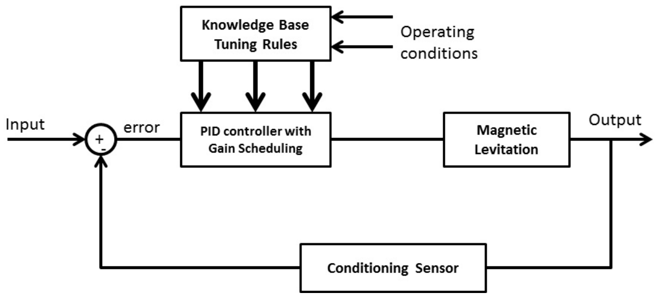


Fig. 4. Gain scheduling block diagram.

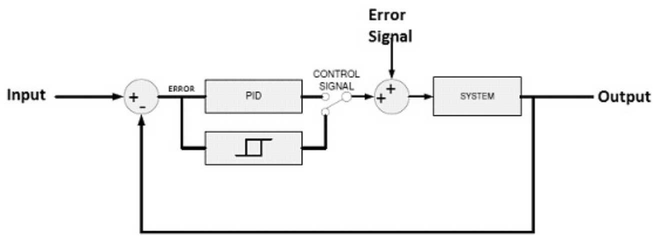


Fig. 5. Relay feedback to obtain PID parameters.

Table 1. Conversion of critical gain and time period to PID parameters.

PID type	K_P	T_i	T_d
P Controller	$0.5K_C$	Inf	0
PI Controller	$0.45K_C$	$T_C/1.2$	0
PID Controller	$0.6K_C$	$T_C/2$	$T_C/8$

selected on the basis of operating conditions of the process with the help of rule defined by operator and specific program.

In earlier days ZN method is used to tune the PID in that method you have to oscillate the system to generate critical time period and critical gain and convert them in the form of PID parameters with the help of ZN table which is a very long process and too much time consuming.

To reduce that time, we want a technique which can tune the PID parameter automatically and can generate best possible response (Table 1). So, for this we are going to use Multilayer Neuro PID.

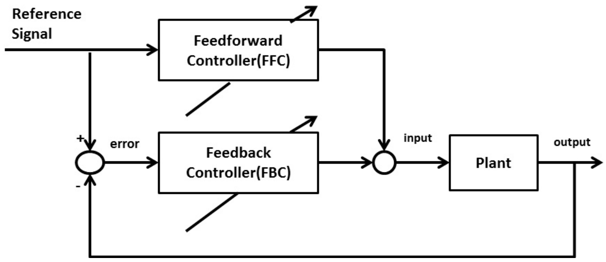


Fig. 6. A general neuro control approach.

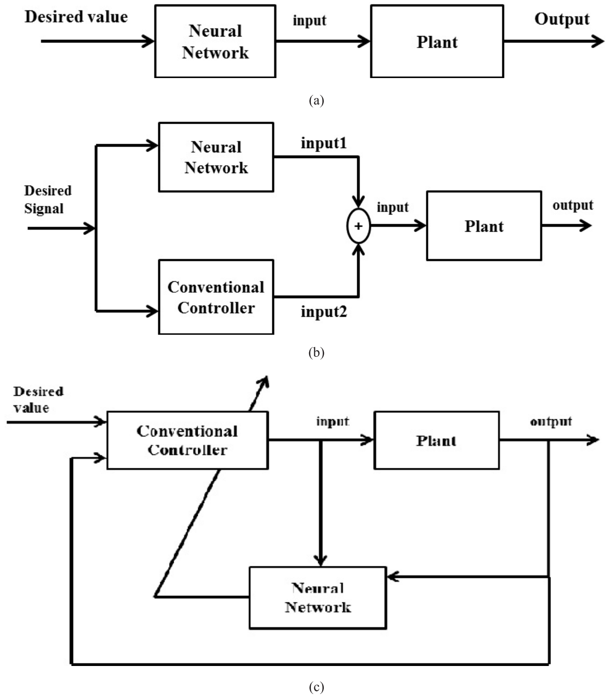


Fig. 7. (a) Series neurocontroller; (b) Parallel neurocontroller; (c) Self-tuning neurocontroller.

5. Neuro Control Approach

In late years, there are so much neural techniques are suggested and neural network is guided to treat as a controller just learning a contrary model of the desired system or also can be accepted as emulator by simple changes in feedforward block. Learning method with Back-propagation are globally used procedures in neural controllers.

The basic controller with back-propagation algorithm as in Fig. 6 using feedforward (FF) and feedback (FB) controllers are adjusted according to the system condition and output. The neural networks are used rather the conventional controller in FF or FB controllers.

According to the neural theory there are basically three classes of neurocontroller:

(a) Series neurocontroller
(b) Parallel neurocontroller
(c) Self-tuning neurocontroller

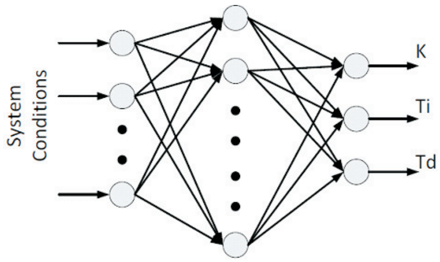


Fig. 8. Multilayer structure for parameters.

The series neurocontroller has inverse model of given plant in the neuro block. The parallel neurocontroller adjusts the control input according the conventional controller of the plant. The self-tuning neurocontroller tunes the parameters of conventional controller on the basis of neural network.

To control and stabilize the magnetic levitation system, we are using series neurocontroller to stabilize the magnetic levitation system (Fig. 7).

6. Implementation and Results

To implement multilayer neural network, we can use single MLP to obtain all the parameter as in the Fig. 8 or we can go for separate MLP for each parameter and to implement the Multilayer neuro PID, we are using separate MLPs for each Parameters K_p, K_i and K_d.

Let us take the transfer function of Magnetic Levitation system

$$G(s) = \frac{77.8421}{0.0311s^2 - 30.5250}$$

For the Magnetic levitation system Simulink model of Multilayer Neuro PID is shown in Fig. 9.

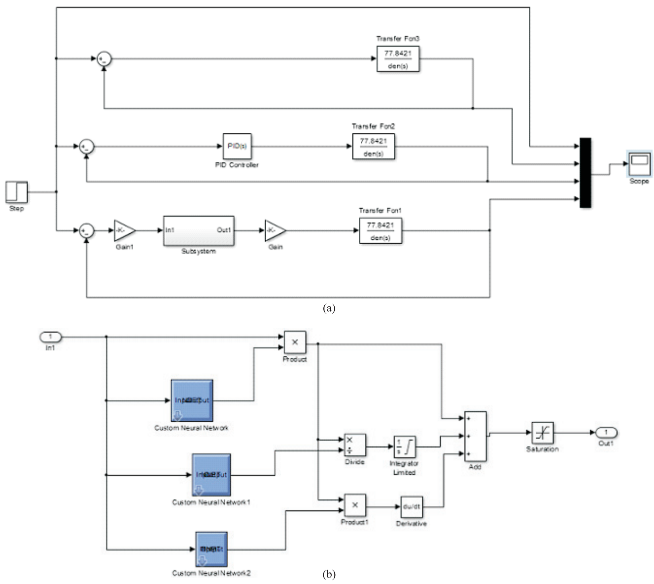


Fig. 9. (a) Simulation of the model; (b) Simulation of neuro PID controller.

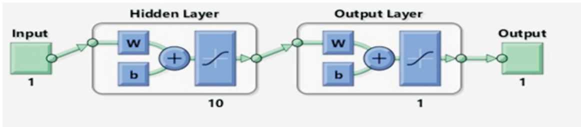


Fig. 10. Training window of neurons.

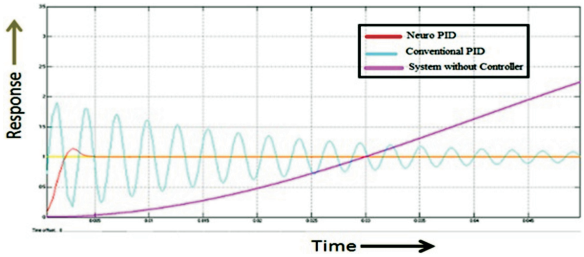


Fig. 11. Step response of magnetic levitation with various controllers.

Table 2. Comparison between various parameters for different controllers.

Parameters	Magnetic levitation without controller	Magnetic levitation with conventional PID	Magnetic levitation with multilayer neuro PID
Steady state error (e_{ss})	NA	1.25	0
Rise Time (t_r)	68 msec	1.2 msec	2.25 msec
Settling Time (t_s)	Never settles	.1195 sec	.0045 sec
Overshoot ($\%M_p$)	230.65	90.48	14.28
Stability	Unstable (Oscillating)	Stable	Stable

We have chosen 10 neurons in hidden layers and 1neuron each in output and input layer as in figure. In the Fig. 10 'w' is synaptic weight and 'b' is bias or threshold value and 'transig' function is chosen as activation function in both layers.

From the Table 2 and Fig. 11, it is clear that system without any controller is highly unstable but with conventional PID, it becomes stable but overshoot & settling time is too high for $K_p = 3$, $K_d = 20$ and $K_i = 0.003$. With the help of Multilayer Neuro PID system becomes stable and settling time and overshoot becomes too small but rise time of the system increase which is a setback but we can neglect this as our basic requirement is less settling time and less overshoot that is achieved as it becomes 4.5 milliseconds and 14.28% respectively.

7. Conclusion

The conventional type of PID controller can also be used in non-linear systems by dividing it into small linear part which is very resilient work. But there is an alternative and easy approach for the auto tuning of PID is Neuro PID. But one thing is mandatory here to point out that this technique is not so much easy especially in the case of continuous controller but we can make it simple by adding programmable controller to control the system without any ease.

The main advantage of Neuro PID controller is that it sustains uniform response for entire operating surface of the plant and it is not affected by the self-parameters of the plant. If some factors affect the response, Neuro PID controller can handle that factors and removes that error to produce uniform response.

References

[1] A. Beric, Artificial Neural Networks for Early Breast Carcinoma Detection, *Proceedings of International Workshop on Neural Networks for Identification Control Robotics and Signal/Image Processing NICRSP-96*, (1996).

[2] E. A. Soliman, Modeling of Microstrip Lines using Neural Networks: Application to the Design and Analysis of Distributed Microstrip Circuits, *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 14, issue 2, pp. 166–173, March (2004).

[3] Sigeru Omatu, Michifumi Yoshioka, Toshihisa Kosaka and Hidekazu Yanagimoto, Neuro-PID Control of Speed and Torque of Electric Vehicle, *International Journal on Advances in Systems and Measurements*, vol. 3, no. 1–2, (2010).

[4] M. H. Moradi, New Techniques for PID Controller Design, In *Conference on Control Applications CCA 2003, Proceedings of 2003 IEEE*, vol. 2, pp. 903–908, June (2003).

[5] Yun Li, Kiam Heong Ang and G. C. Y. Chong, PID Control System Analysis and Design, *Control Systems, IEEE*, vol. 26, issue 1, pp. 32–41, February (2006).