Proceedings of the International Conference on Man-Machine Systems (ICoMMS)
11 – 13 October 2009, Batu Ferringhi, Penang, MALAYSIA

Neural-tuned PID Controller for Point-to-point (PTP) Positioning System: Model Reference Approach

Wahyudi, Wali Ahmad (a) Myo Min Htut
Intelligent Mechatronics Research Unit
Department of Mechatronics Engineering, Faculty of Engineering
International Islamic University Malaysia (IIUM), PO BOX 10, 50728, Kuala Lumpur, Malaysia
wahyudi@iiu.edu.my

Abstract: **Point-to-Point (PTP) motion control systems play an important role in industrial engineering applications such as advanced manufacturing systems, semiconductor manufacturing systems and robot systems. Until now, Proportional-Integral-Derivative (PID) controllers are still the most popular controller used in industrial control systems including PTP motion control systems due to their simplicity and also satisfactory performances. However, since the PID controller is developed based on the linear control theory, the controller gives inconsistent performance for different condition due to system nonlinearities. In order to overcome this problem, Neural-tuned PID control using Model Reference Adaptive Control (MRAC) is proposed. By using Extended Minimal Resource Allocation Algorithm (EMRAN) to train the Radial Basis Funciton (RBF) Network, the PID controller can learn, adapt and change its parameters based on the condition of the controlled-objectin real-time. The effectiveness of the proposed method is evaluated experimentally in real time using an experimental rotary positioning system. The experimental results show that the proposed system is better than classical PID controller in terms of not only positioning performance but also robustness to inertia variations.**

I. INTRODUCTION

Applications of Motion control can be seen in various engineering files such as automation, robotics, manufacturing, production, precision control, bioengineering, and semi-conductor industries. Basically there two kinds of motion control systems namely point-to-point (PTP) positioning systems and continuous path (CP) control systems [1]. Both motion control systems generally need a good controller to realize high speed and high precision motion system. However, it is not easy to achieve high precision system because of non-linearities such as friction and saturation exist in the motion control systems. However, since both friction and saturation are non-linear function, friction and saturation can not be compensated effectively by controller designed based on the linear control theory.

The motion control systems are also characterized by parameter uncertainties. Two major sources of the parameter variations in positioning systems are inertia and friction variations. Inertia of the positioning systems may vary due to payload variation. Friction variation may occur due to variations of the lubrication condition and/or inertia. Inertia variation can cause variation of the Coulomb friction as well, as in [2]. Therefore, the robustness of the control system is also an important requirement of the high performance motion control systems.

Although many innovative methodologies have been devised in the past 50 years to handle more complex control problems and to achieve better performances, the vast majority of industrial processes, not less than 90% are still controlled by means of simple Proportional-Integral-Derivative (PID) controllers as estimated in [3]. Until now, PID controllers are still the most popular controller used in industrial control systems including motion control systems due to their simplicity and also satisfactory performances [4]. The use of PID controller for motion control systems was proposed by many researchers, for example discussed in [5-6]. However, since the PID controller is developed based on the linear control theory, the controller gives inconsistent performances for different condition as discussed in [7]. Improvement of the PID controller for achieving high performance (especially high speed and high precision) of motion control system is required.

To overcome the above-mentioned problem, in this paper, intelligent PID controller based on the artificial neural network is introduced for the PTP positioning system application so that the PID controller can learn, adapt and change its parameters based on the condition of the controlled-object. The neural network combining with PID control can partly solve the problem of PID controller parameter's real-time tuning [8]. Network tuner extracts tuning knowledge automatically through the use of a representative process, and therefore knowledge extraction from a human control [9]. Most of the research on PID self-tuning has been done using back propagation network. Traditional BP neural network has the disadvantages of low approximating ability and convergence property [10]. But the Radial Basis Function (RBF) neural network can approximate any continuous functions at any precision, and need shorter duration to train than back propagation [11].

In the proposed intelligent controller, an Extended Minimal Resource Allocation Algorithm Network (EMRAN) with Model Reference Adaptive Control (MRAC) is adopted. EMRAN that is well suited for real-time implementation of nonlinear plants was proposed in [12]. EMRAN, a powerful variation of the standard Minimal Resource Allocation Network (MRAN), is applied to train the neural network in this research. MRAN is a sequential learning technique for RBF neural networks [13].

In the proposed controller, firstly the nominal PID controller parameters are designed based on the nominal plant parameters using any standard controller design, then the EMRAN is included and then trained so as the PID controller parameters can adapt and change due to plant parameter variations. The effectiveness of the proposed intelligent PID controller is evaluated experimentally on experimental rotary motion system. The experimental results confirm the effectiveness of the proposed controller in term of performance and robustness to inertia variation in comparison with the normal PID controller.

II. SYSTEM DESCRIPTION

The experimental rotary positioning system considered in this paper is the Quanser Servo plant as shown in Fig.1. The plant consists of servo motor driven by a power amplifier and encoder as the feedback sensor. US Digital Optical Kit Encoder is used to measure the load shaft angular position. It offers high resolution of 0.088 deg, and measures the relative angle of the shaft. The controller is implemented digitally on a PC using MATLAB-based Wincon software. Both the controller signal is sent to controlled object and the encoder signal is fed back to controller using Quanser MultiQ PCI data acquisition system. In addition, the system facilitates additional loads to alter the inertia for robustness evaluation. The mathematical model of the experimental rotary positioning system was developed by applying physical laws and neglecting all the system nonlinearities. The developed dynamics model of the plant is described in (1).

$$\frac{\theta_l(s)}{V_m(s)} = \frac{64.12}{s(s+36.43)} \qquad (1)$$

Fig. 1. Experimental Rotary Positioning System

III. PROPOSED NEURAL-TUNED PID CONTROLLER

A. Structure of Neural- PID with Model Reference

Model reference adaptive control (MRAC) systems are used to solve problems where the specifications are given in terms of a reference model. The reference model describes how the process output ideally should respond to the command signal. The adaptation algorithm attempts to adjust the parameters such that the adjustable system correlates with the reference model. In MRAC systems, the adaptation process adjusts the system's parameters in such a way t hat the error between the process output and the reference output is minimized. The key problem is to determine the adaptation algorithm such that a stable system, that brings the error to minimal, is obtained. The structure of the EMRAN with MRAC is shown in Fig. 2.

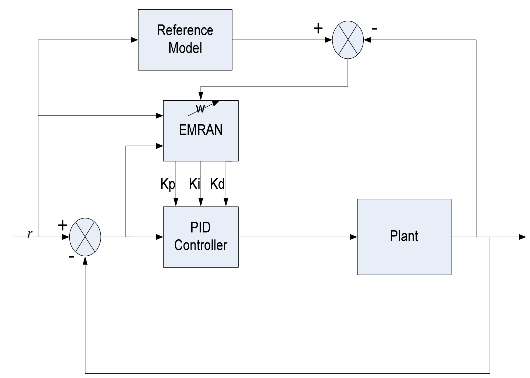

Fig. 2. Structure of EMRAN-PID with Model Reference

The following specifications are set in calculating the second order Reference Model based on the standard second order model as in (2)

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad (2)$$

The following criteria are set as the desired response of the reference model

Percentage Overshoot =2%
Settling Time = 0.3s
Error = 0.01

Input vectors to neural network are taken from Reference angle (ref) and error (e). The difference between the reference model output and plant output is taken as the error in training the neural system. Basically the controller is a PID controller with standard controller parameter gains of K_p, K_i and K_d. However, in the proposed controller, an EMRAN neural tuner with model reference control is added so that the PID controller parameters (K_p, K_i and K_d) change in real time due to variation of the error and reference input conditions. The proposed Neural-tuned PID controller is designed by using the following two simple procedures:

1. Design nominal values of PID controller gains based on the linear model of the plant and desired performance.
2. Design and train EMRAN-based neural tuners for K_p, K_i and K_d to search the best controller gains, based on the nominal values of PID controller gains error conditions.

B. Nominal PID Control Design

The PID controller for position control is mathematically represented as

$$G_c(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \dot{e}(t) \qquad (3)$$

where K_p, K_i and K_d are proportional, integral, and derivational gains respectively. The PID controller gains are tuned based on the linear model of the plant using Simulink Response Optimization toolbox. In order to realize the desired response, the following specifications are set

Settling time = 0.3s
Overshoot = 2%
Steady-state error = 0.01

Finally, the optimized values of the PID controller gains are K_p=20, K_d = 0.3 and K_i=0.1.

C. Design of Extended Minimal Resource Allocation Network

The extended minimal resource allocation network (EMRAN) uses Gaussian functions as the activation functions which are based on the radial basis function neural network architecture. With an initial of no hidden neurons EMRAN will allocate the minimal number of hidden neurons required for the neural network for system identification and control. Fig. 3 shows the structure of an RBF neural network with Gaussian Function [14]. The illustrated network has *n* number of input and output, i.e. *x* and *y* respectively and *N* number of hidden neurons.

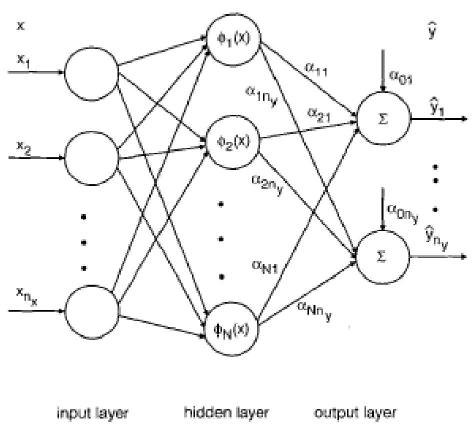

Fig. 3. Radial Basis Function Neural Network

The output of this neural network can be described by the following relationship as in (4)

$$f(x) = \alpha_0 + \sum_{n=1}^{N} \alpha_n \phi_n(x) \qquad (4)$$

where, α_o is the bias, α_n is the connecting weight and ϕ_n is the response of the *n*th hidden unit to the input *x*. The EMRAN algorithm which is developed in [12] is trained online and the parameters i.e. weight, width and centre are also adjusted online using the Extended Kalman Filter (EKF) scheme. The learning process of EMRAN involves the addition of hidden neurons, pruning of hidden neurons and adopting the winner-takes-all algorithm to identify the winning neuron to be updated.

For each set of input-output pair, (x_i,y_i) the hidden neuron is added once the all of these criteria are satisfied

$$\left\| x_i - \mu^i_{nr} \right\| > \varepsilon_1 \qquad (5)$$

$$e_i = \left\| y_i - f(x_i) \right\| > \varepsilon_2 \qquad (6)$$

$$\sqrt{\frac{\sum_{j=i-(n_w-1)}^{i} e_j^T e_j}{n_w}} > \varepsilon_3 \qquad (7)$$

where, ε_1, ε_2 and ε_3 are the threshold to be set by the user and μ^i_{nr} is the centre of the Gaussian function. Equation (5) indicates that the hidden neurons will be added if the distance of the input is far away from the threshold, ε_1. If the error specification as in (6) is satisfied i.e. larger than, ε_2, then finally the sum squared error i.e. (7) is checked to confirm that it is large than ε_3.

Once all of these criteria are met only then a hidden neuron will be added. If these criteria are not met then the centre, width and weight are updated using EKF which can be illustrated as follows

$$\alpha^i_{h+1} = e_i \qquad (8)$$

$$\mu^i_{h+1} = x_i \qquad (9)$$

$$\sigma^i_{h+1} = \kappa \left\| x_i - \mu^i_{nr} \right\| \qquad (10)$$

where, α^i_{h+1} is the connecting weight vector, σ^i_{h+1} is width of the Gaussian function and κ is the overlap factor. Once hidden neurons are added the algorithm will perform the pruning strategy to eliminate the neurons which are not computationally effective. The criterion for elimination is based on the following steps

- Compute the output of all the hidden units, $\mathbf{o}^i_k$
- Obtain the largest output from the among the output of all the hidden units, $\mathbf{o}^i_{max}$

Compute the normalized output for each of the hidden unit

$$\text{i.e. } r_k^i \quad \frac{\left\| o_k^i \right\|}{\left\| o_{\max}^i \right\|} \tag{11}$$

Compare the value of the normalized output, r_k^i for every hidden unit with a threshold, and remove the hidden neuron which has $r_k^i <$.

Update the dimensionality of the network using EKF to suit the reduced network

By adopting the "winner-takes- all" algorithm only the winning weight, centre and width of the hidden neuron will be updated and thus this will make the algorithm less computationally expensive.

D. Training the EMRAN Network

Firstly, EMRAN outputs are scaled based on the nominal value of PID. For proportional gain K_p, the nominal gain of 20 is used as a mean value, resulting the scaling range from 10 to 30. For integral gain K_i, the EMRAN outputs are scaled to be from nominal value of 0.1 to three times of it, which is to 0.3. For derivative gain Kd, the EMRAN outputs are scaled to be from 50% less of nominal Kd, which is 0.15, to the three times of nominal Kd which is 0.9. The range is selected so that artificial neural network can find a best value near the region of nominal PID gains, depending on the condition of the controlled object. Secondly, the EMRAN tuners were trained online using a stair case input ranging at every 20-deg intervals so that the network will learn the process at every 20-deg intervals.

The learning process of EMRAN involves the addition of hidden neurons, pruning of hidden neurons and adopting the winner-takes-all algorithm to identify the winning neuron to be updated. Thirdly, after the desired performance is met i.e. when the plant output following the training signal according to the desired performance, the last set of weights is saved. EMRAN tuners will be adjusting the gains of PID depending on the error of the plant and the reference angles, based on the set of weights trained online during learning process.

IV. RESULTS

The effectiveness of the proposed Neural-tuned PID controller is evaluated experimentally with the rotary positioning system and is compared with that of the classical PID controller which uses the nominal values of the gains obtained by using Matlab Simulink Optimization Toolbox. Figs 4 and 6 show the responses to 10-deg and 90-deg step input when the controller is evaluated on the nominal plant. The positioning performances in terms of overshoot, settling time and positioning accuracy are listed in Table 1. As shown in Figs. 4, 6 and Table 1, the proposed Neural-tuned PID controller gives better positioning performances in terms of steady-state error, and settling time while giving slightly higher percentage overshoot.

Furthermore, the robustness of the proposed Neural-tuned PID controller is evaluated experimentally by adding additional load to the rotary system. Figs 5 and 7 show the responses to 10-deg and 90-deg step inputs when the controller is implemented on the increased inertia plant. When the inertia is changed, EMRAN-PID with Model reference control gives better positioning performance than PID in all three evaluation criterion of steady-state error, settling time and percentage overshoot. The positioning performances in terms of overshoot (PO), settling time (T_s) and positioning accuracy are also listed in Table 1. As shown in Table 1, the positioning performances do not change significantly even if the inertia becomes larger. Hence it can be concluded that the proposed Neural-tuned PID controller is more robust to inertia variation than the classical PID controller.

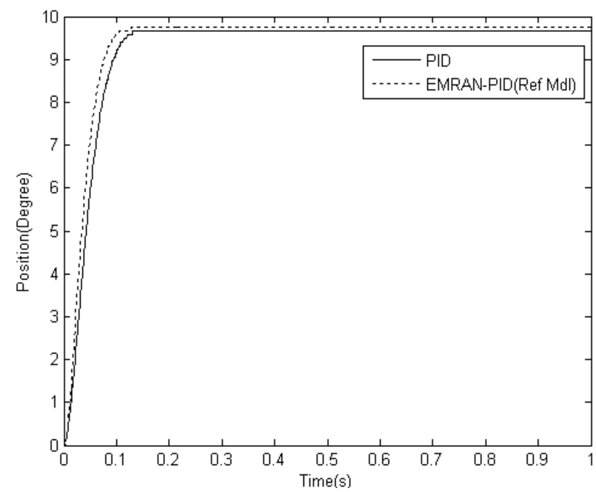

Fig. 4. Response to 10-deg step input on Nominal Plant

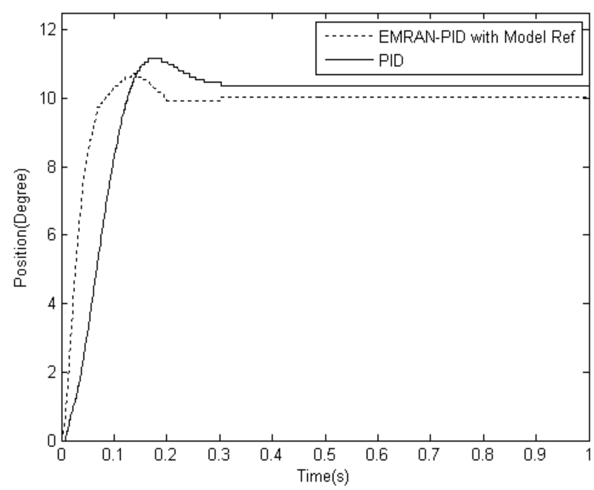

Fig .5. Response to 10-deg step input on increased inertial Plant

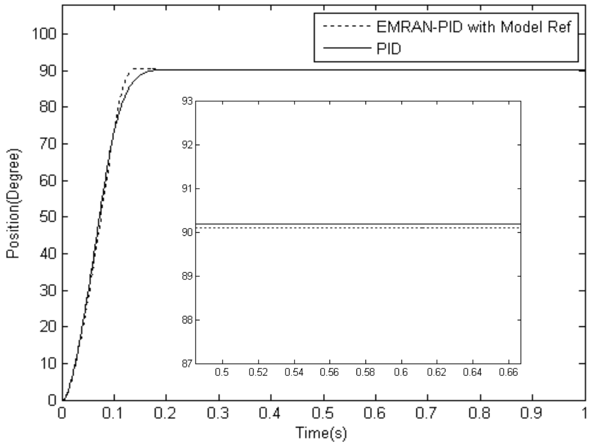

Fig. 6 Response to 90-deg step input on Nominal Plant

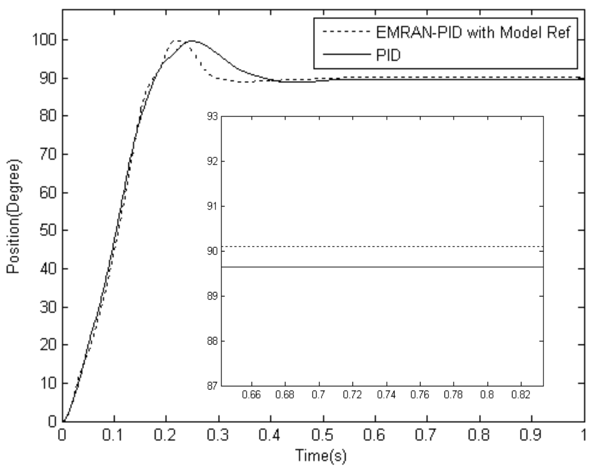

Fig .7 Response to 90-deg step input on increased inertial Plant

TABLE 1
PERFORMANCE COMPARISON

Reference angle	Plant	Controller	%PO	T_s (seconds)	Accuracy (Error)
10-deg	Nominal	PID	0	0.132	0.332
		EMRAN-PID	0.88	0.08	0.02
	Increased Inertia	PID	7.63	0.34	0.3759
		EMRAN-PID	7.96	0.32	0.07
90-deg	Nominal	PID	0	0.143	0.1759
		EMRAN-PID	0.59	0.208	0.0879
	Increased Inertia	PID	11.08	0.56	0.2673
		EMRAN-PID	10.54	0.541	0.0879

V. CONCLUSION

This paper has introduced the neural-tuned PID controller using EMRAN with Model Reference control for PTP positioning system. First, the nominal values PID controller is designed based on the desired transient performance. Next, the EMRAN tuners are designed so that the PID controller changes its gains automatically due to error and reference input conditions. The effectiveness of the proposed controller is evaluated experimentally to control the experimental rotary positioning system and is compared with the classical PID controller. The results confirm that the proposed Neural-tuned PID controller with Model Reference control is not only more effective but also more robust to inertia variation than the classical PID controller.

ACKNOWLEDGMENT

This research is financially supported by Research Management Centre under IIUM Fundamental Research Grant Scheme IFRG 0702-61.

REFERENCES

[1] Crowder R.M, *Electric Drives and their Controls*, Oxford, Oxford University Press, 1998.

[2] Armstrong-Helouvry B, Dupont P, and De Wit C, "A survey of models, analysis tools and compensation method for the control of machines with friction," Automatica, Vol. 30, No. 7 (1994) pp. 1083-1138, 1994.

[3] P.B. Deshpande, Improved quality control online with PID Controllers, Chem, Eng. Progress, pp. 71-76, 1991

[4] Johnson M. A, and Morad M. H, *PID Control, New Identification & Design Methods*, Springer-Verlag London Limited, 2005.

[5] Yonezawa H, Hirata H, and Sasai H, "Positioning table with high accuracy and speed," Annal CIRP, Vol. 39, pp. 433-436, 1990.

[6] Sato K, Shinshi T, Abidin Z, and Shimokohbe A, "Positioning performance of a leadscrew system with six kinds of control methods - basic positioning performance and effect of Coulomb friction on the performance," Proceedings of China-Japan Bilateral Symposium on Advanced Manufacturing Engineering (Yellow Mountain, China), 1998, pp. 39-44.

[7] Chao C, Neou J and Wang H, "Model reference adaptive control of long stroke positioning systems with piecewise linear gain," International Journal of JSPE Vol. 32, No. 3, pp. 200-206, 1993.

[8] Zhenhai G, and Bo Z, (2005). "Vehicle lane keeping of adaptive PID control with BP neural network self-tuning," IEEE.

[9] Chan K.C, Leong S.S, and Lin I.C.G, (1995). "A neural network PI controller tuner," Artificial Intelligence in Engineering, (167-176).

[10] Yu W.H, and Lu T.C.L, (1998). "PID controller using Dynamic Neural Networks," IEEE

[11] Shi T, Xia C, Wang M, and Zhang Q, (2006). "Single Neural PID Control for Sensorless Switched Reluctance Motor Based on RBF Neural Network," Proceedings of 6th World Congress on Intelligent Control and Automation, Dalian, china.

[12] Lu Y, Sundararajan N, and Saratchandran P, "Analysis of Minimal Radial Basis Function Network Algorithm for Real-time identification of nonlinear dynamic systems." IEEE Proceedings on Control theory and Application. 2000; Vol.4, no. 147, pp.476.

[13] Platt J.C, (1991). "A resource allocating network for function interpolation," Neural Computing., 5, pp. 595-603.

[14] Fravolini M. L, Campa G, Napolitano M, and Cava M.L, "Comparison of Different Growing Radial Basis Functions Algorithms for Control Systems Applications," Proceedings of the American Control Conference, Anchorage, AK May 8-1 0, 2002.

[15] Campa G, Fravolini M. L, and Napolitano M, "A library of Adaptive neural networks for control purposes," IEEE International Symposium on Computer Aided Control System Proceedings. Sep 18-20, Glasgow, Scotland, UK, 2002.