Comparison of Swarm Adaptive Neural Network Control of a Coupled Tank Liquid Level System

M.S.Ramli, M.A. Ahmad, R.M.T. Raja Ismail,
Control and Instrumentation Research Group (COINS)
Faculty of Electrical & Electronics Engineering, University Malaysia Pahang (UMP)
Lebuh Raya Tun Razak, 26300, Kuantan, Pahang, Malaysia
{syakirin, mashraf, rajamohd }@ump.edu.my

***Abstract* – This paper presents the use of neural network control approaches for a two inputs –two outputs (TITO) coupled tank liquid levels with disturbances effects and set-point changes in dynamic system. Hybrid PI-neural network (hybrid PI-NN) and PID neural network (PID-NN) controllers are the techniques used in this investigation to actively control the liquid levels of coupled tank system. Unlike traditional neural network weight adaptation using gradient descent method, Particles Swarm Optimization (PSO) is utilized for adaptive tuning of neural network weights adjustment and fine tuning the controller's parameters. A complete analysis of simulation results for each technique is presented in time domain. Performances of both controllers are examined in terms of disturbance rejection and control performance measures for common input changes. Finally, a comparative assessment on the impact of each controller on the system performance is presented and discussed.**

***Key-words:* NN, PSO, level control, water tank**

I. INTRODUCTION

Process control techniques always deal with increasing production rate and product quality of industrial enterprises while keeping the costs as low as possible. In process control, precise liquid level control of storage tanks and reaction vessels is essential in many industrial operations and mainly in chemical engineering systems where the liquids are pumped to the tanks, stored and flow through the coupled tanks [1]. Namely, the coupled tank system has wide applications in many commercial and industrial sectors. It is most commonly used in the area of water purification, chemical and biochemical processing, automatic liquid dispensing, food and beverage processing [2], and pharmaceutical industries [3]. The coupled tank system has two vertical tanks joined with an orifice and has inlet liquid pumps and discharge valves.

Various attempts in controlling the liquid levels of a coupled tank system were proposed. For example, sliding mode control strategy was applied to the coupled tank system such as discussed in [4]. However, this technique is only best applied on a Single-Input-Single-Output (SISO) system. Another control strategy is Proportional Integral (PI) with decoupling controllers [5]. The technique presented in the paper utilizing the root locus approach in determining the controllers' parameters. Besides that, there were several other control strategies investigated in the literatures [6],[7],[8].

The particle swarm optimization (PSO), first introduced by Kennedy and Eberhart [9], is an evolutionary computation technique that is initialized with a population of random solutions. The most prominent merit of PSO is its fast convergence speed. In addition, PSO algorithm can be realized simply for less parameters need adjusting. Now it was applied successfully in many areas such as in controller's optimization [10], system identification [11], and solving optimal power plant problem [12].

Accurate model and its parameters which capture the characteristic of the coupled tank system are required for designing its controller for achieving a good performance. The specific point tackled in the paper is about the advantages of using a new hybrid PI-NN instead of a PID-NN controller, consisting of PSO, Neural Network (NN) and PI controller. From a generic tuning rule the optimum settings from an Integral Squared Error and Integral Time Absolute Error criterion point of view are derived. The validation result shows that hybrid PI-NN controller much faster than PID-NN and also good robustness and small steady-state error.

II. PROCESS PLANT DESCRIPTION

The schematic diagram of the coupled tank system considered in this work is shown in Fig.1 below where Q_i= $\{Q_{i1}, Q_{i2}\}$ are the inlet flow rate to tank 1 and tank 2, Q_{12} is the liquid flow rate from tank 1 to tank 2 through orifice, Q_o= $\{Q_{o1}, Q_{o2}\}$ are the outlet flow rate of tank 1 and tank 2, and h= $\{h_1, h_2\}$ denotes the liquid level of tank 1 and tank 2, respectively. In this simulation, the target is to control the level in two tanks by the inlet liquid flow from two pumps. The process input are u= $\{u_1(t), u_2(t)\}$ (voltage input to pumps) and the output are h= $\{h_1(t), h_2(t)\}$ liquid level in tank 1 and tank 2 respectively.

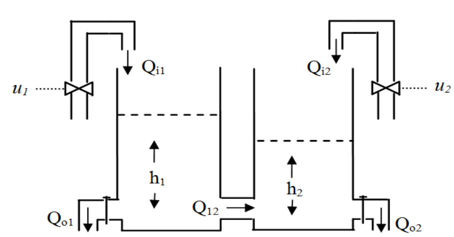

Figure 1. Schematic of Coupled Tank Process

978-0-7695-3892-1/09 $26.00 © 2009 IEEE
DOI 10.1109/ICCTD.2009.124

The nonlinear plant equations can be obtained by mass balance equations and Bernoulli's law. After linearization process, the linear plant equations can be obtained as:

$$\dot{h}_1(t) = \frac{k_1}{A}U_1(t) - \frac{\beta_1 a}{A}\sqrt{\frac{g}{2\bar{h}_1}}H_1(t) + \frac{\beta_x a}{A}\sqrt{\frac{g}{2|\bar{h}_2 - \bar{h}_1|[H_2(t) - H_1(t)]}}$$
$$\dot{h}_2(t) = \frac{k_2}{A}U_2(t) - \frac{\beta_2 a}{A}\sqrt{\frac{g}{2\bar{h}_2}}H_2(t) + \frac{\beta_x a}{A}\sqrt{\frac{g}{2|\bar{h}_2 - \bar{h}_1|[H_2(t) - H_1(t)]}} \qquad (1)$$

where A is the cross sectional area of tank 1 and tank 2 (cm^2), a is the cross sectional area of outlet hole of tank 1, tank 2 and the cross sectional area of jointed opening between tank 1 and tank 2 (cm^2),β_1 is the valve ratio at the outlet of tank 1, β_2 is the valve ratio at the outlet of tank 2, β_x is the valve ratio between tank 1 and tank 2, $\bar{h}_1$, $\bar{h}_2$ are the steady-state water level of tank 1 and tank 2, g is the gravity (cm^2/s) and k_1, k_2 are the gain constants of pump 1 and pump 2 (cm^3/V.s), respectively.

From the linear plant equations (1), it can be transformed to yield a nominal block transfer function of the form (2)

$$\begin{bmatrix} h_1(s) \\ h_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \qquad (2)$$

Through simple algebraic manipulation, the transfer matrix $G_{ij}(s)$ yields to

$$\left.\begin{aligned} G_{11}(s) &= \frac{\frac{k_1}{A}\left(s + \frac{T_x + T_2}{T_2 T_x}\right)}{\Delta} \qquad G_{12}(s) = \frac{\frac{k_2}{A}\left(\frac{1}{T_x}\right)}{\Delta} \\ G_{21}(s) &= \frac{\frac{k_1}{A}\left(\frac{1}{T_x}\right)}{\Delta} \qquad G_{22}(s) = \frac{\frac{k_2}{A}\left(s + \frac{T_x + T_2}{T_1 T_x}\right)}{\Delta} \end{aligned}\right\} \qquad (3)$$

$$\Delta = s^2 + \left(\frac{T_1 T_x + T_2 T_x + 2T_1 T_x}{T_1 T_2 T_x}\right)s + \left(\frac{1}{T_1 T_2} + \frac{1}{T_1 T_x} + \frac{1}{T_2 T_x}\right)$$

provided that T_1 is the time constant of tank 1, T_2 is the time constant of tank 2 and T_x is the time constant interaction between tank 1 and tank 2.

According to transfer matrix $G_{ij}(s)$ in (2) and (3), the transfer functions of coupled-tank process are second order form which have cross coupling between process input and outputs. The decoupling controllers are required for minimizing the effects from cross coupling and transform TITO plant transfer function into SISO form. This is where neural network structure is introduced at which can be functioning as the de-coupler controller.

III. HYBRID PI-NEURAL NETWORK CONTROLLER

A combinational PI controller with neural network structure for controlling the liquid level system of the coupled tank is presented here. Proportional-Integral (PI) controller is a feedback controller which drives the plant to be controlled with a weighted sum of error (difference between output and desired response) and the integral of that value. The general model for a PI controller is given in (4) where H_k is the process variable, E_k is the difference between the output and the desired response, K_{Pk} and K_{Ik} are the proportional and integral gains respectively.

$$G_{PI_k}(s) = \frac{H_k(s)}{E_k(s)} = \frac{sK_{P_k} + K_{I_k}}{s} \qquad (4)$$

The hybrid PI-NN is constructed by series cascading the PI controllers with a neural network structure as shown in Fig. 2. Throughout the network, the linear activation function is used in all neurons.

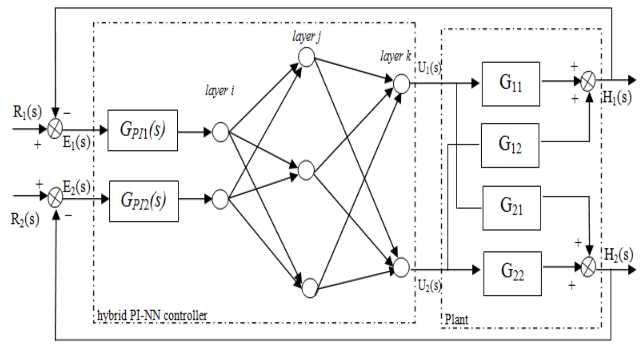

Figure 2. (TITO) process with hybrid PI-NN control system

Fig. 2 shows the plant transfer function $G_{ij}(s)$ that has the cross coupling between process inputs and outputs. Because of interaction between processes, the neural network structures will basically act as a de-coupler controller for minimizing the cross coupling effects via its connective weight adaptation.

For hybrid PI-NN controller, the manipulated variable signals injected into the plant is obtained as

$$U_k = O_k \qquad (5)$$

where we have

$$O_k = \sum_j W_{kj} O_j \text{ and } O_j = \sum_i W_{ji} O_i$$

The net output O_i on the other hand, comes from equation (4) which yield to $O_i = G_{PI_k}(s)E_k(s)$

IV. PID NEURAL NETWORK CONTROLLER

A control structure for controlling the liquid level tank using PID neural network controller as shown in Fig.3 with input of s_j and output θ_j. The property of a neuron is decided by the input-output activation function (f) whereby the P-neuron, I-neuron and D-neuron are representing the proportional (P), integral (I) and derivatives (D) functions, respectively.

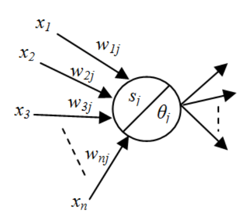

Figure 3. .neuron form

For any neuron (namely the j^{th} neuron) in the network which has n-1 inputs, at any t time, the input of the neuron is given by

$$s_j(t) = \sum_{i=1}^{n-1} w_{ij} x_i(t) \quad (6)$$

where $x_i(t)$ are the outputs of n-1 connected neurons in foregoing layer and w_{ij} are the connected weights. The output of this neuron will depend on its activation function which can be proportional (P), integral (I) or derivative (D) functions.

TABLE I. ACTIVATION FUNCTION FOR EACH TYPE OF NEURON

Type of neuron	Output, $\theta_j(t)$
P	$s_j(t)$
I	$\int_0^t s_j(t)dt$
D	$\frac{ds_j(t)}{dt}$

Basic PID-NN consists of two input neurons and one output neuron whereby the hidden layer of this network structure is made of three neurons which each representing P, I and D activation function respectively. Their output's relationship for each function is summarized in Table 1 above. Fig. 4 below shows the fundamental network layer of a basic PID-NN.

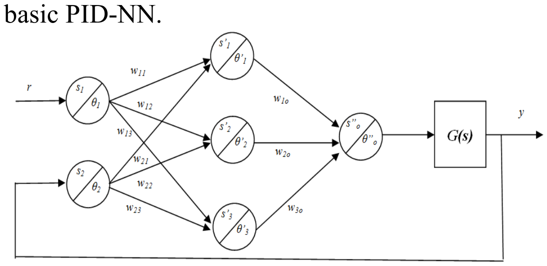

Figure 4. Basic PID-NN

Through connective weight adaptation between layers, the PID-NN is actually acting as a conventional PID controller. Since PID controllers have been widely used in industry, that is to say there are much experience to choose P, I and D parameters in order to suit the system's stability without changing one's plant.

The control system for controlling the coupled tank level system consists of several basic PID-NN whereby every basic PID-NN is a sub-net. The multi PID-NN control system is shown in Fig. 5 below.

The structure of multi PID-NN is special. If suitable connective weights are obtained, each sub-net of PID-NN is comparatively equal to a PID controller. By referring to Fig.3 of the basic PID-NN, let say that $w_{1j} = +1$, $w_{2j} = -1$, $w'_{1o} = K_P$, $w'_{2o} = K_I$, $w'_{3o} = K_D$,Then, the input to the network structure will be

$$s'_j = \sum_i w_{ij} x_i = r - y = e \quad (7)$$

Meanwhile, the network output (depending on the type of neuron) of the hidden layer for each neuron is obtained as

$$\theta'_j = f(s'_j) \quad (8)$$

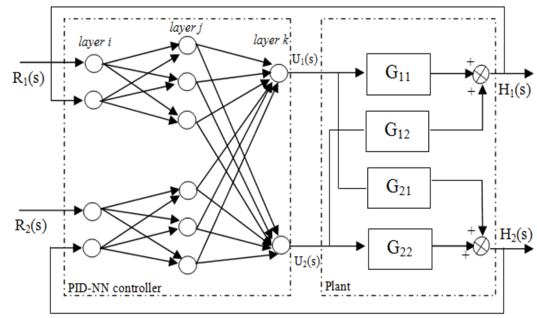

Figure 5. (TITO) process with multi PID-NN control system

Therefore, we derived the total output for the basic PID-NN as shown in Fig.3 as

$$\theta''_o = \sum w'_{jo} x'_j = \left[K_P e + K_I \int edt + K_D \, {de}/{dt}\right] \quad (9)$$

At any rate, the manipulated variable signals injected into the plant as shown in Fig.5 is obtained as

$$U_k = \theta''_k \quad (10)$$

V. PARTICLES SWARM OPTIMIZATION

A. Principles of PSO

The particle swarm optimization (PSO), first introduced by Kennedy and Eberhart [9], is an evolutionary computation technique that is initialized with a population of random solutions. Its development was based on observations of the social behaviour of animals such as bird flocking, fish schooling, and swarm theory. This technique has been widely used in across wide range of application such as in communication, biomedical [13]. It also has, very recently, emerges as an important combinatorial metaheuristic technique for both continuous-time and discrete-time optimization.

In past several years, PSO algorithms have been successfully applied in many research and application areas. It has been demonstrated that PSO gets better results in a faster and cheaper way as compared with other methods. Like other evolutionary computations, PSO can typically initialize a pool of particles with random bird positions (called agent) in two-dimensional space [13] where each is represented by a point in the X-Y coordinates, and the velocity is similarly defined. Bird flocking is assumed to optimize a certain fitness function. The system is initialized with a population of random solutions and searches for optima by updating generations. Instead of using evolutionary operators to manipulate the potential solution (particle), each particle (denoted by ($x_i = (x_{i1}, \cdots, x_{iD})^T$) in PSO flies in the search space with velocity

($v_i = (v_{i1}, \cdots, v_{iD})^T$) which is dynamically adjusted according to its own flying experience and its companions' flying experience. Each agent knows its best value so far (*pbest*) and its current position. This information is an analogy of personal experience of an agent. Each agent tries to modify its position using the concept of velocity. The velocity of each agent can be updated by the following equation:

$$v_i^{n+1} = \omega v_i^n + \eta_1 \Gamma_1 \left(pbest_i - x_i^n\right) + \eta_2 \Gamma_2 \left(gbest - x_i^n\right) \tag{11}$$

where v_i^n is the velocity of agent i at iteration n, ω is weighting function, η_1 and η_2 are weighting factors, Γ_1 and Γ_2 are the cognitive and social learning parameters which generated randomly between 0 and 1, x_i^n is the current position of agent i at iteration n, $pbest_i$ is the *pbest* of agent i, and *gbest* is the best value so far in the group among the *pbest* of all agents. The following weighting function is normally used in equation (11):

$$\omega = \omega_{\max} - \left(\frac{\omega_{\max} - \omega_{\min}}{iter_{\max}} \right) \times iter \tag{12}$$

where $\omega_{\max}$ is the initial weight, $\omega_{\min}$ is the final weight, $iter_{\max}$ is the maximum iteration number, and $iter$ is the current iteration number. Using the previous equation, a certain velocity, which gradually brings the agent close to *pbest* and *gbest*, can be calculated. The current position (search point in the solution space) can be modified by the following equation:

$$x_i^{n+1} = x_i^n + v_i^n \tag{13}$$

B. Model Reference adaptive tuning using PSO

In both hybrid PI-NN and PID-NN control systems; the aim of the controllers' algorithm is to minimize the following fitness function (F_t):

$$F_t = \sum_{k=1}^{z} E_k^{\,2} = \frac{1}{m} \sum_{k=1}^{z} \sum_{q=1}^{m} \left[\{ G_{ref}^k R_k \}(q) - H_k(q) \right]^2 \tag{14}$$

where k is the signal channel number, R_k is the desired set-points and H_k is the outputs of the system as shown in Fig.2 and Fig.5. Meanwhile, q *(=1,2,…,n)* is the serial number. G_{ref}^k is the first order model reference transfer function and is represented as:

$$G_{ref}^k = \frac{1}{\tau_k s + 1} \tag{15}$$

where τ_k is the time constant for shaping the output transient responses to be as desired. Fig.6 below shows the block diagram of model reference adaptive tuning. The weights adjuster block adaptively changes the controller's weights as such the fitness function in (14) is minimized.

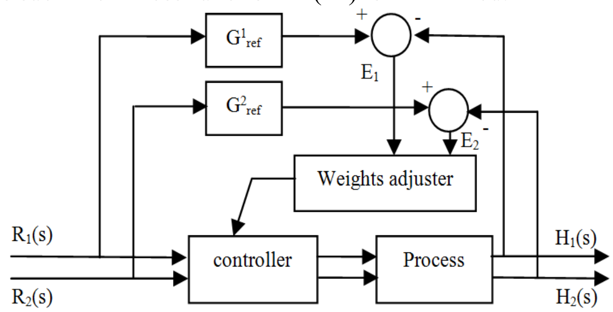

Figure 6. Block diagram of model reference adaptive tuning

The connective weights of PID-NN and hybrid PI-NN as well as the PI parameters are changed and optimized on each iteration n of the PSO. Before beginning the optimization, a population size (i.e number of particles) N and a maximum number of iterations $iter_{\max}$ are chosen. The computation flow of PSO technique can be described in the following steps.

Step 1. Randomly initialize the population: select the (normalized) particle positions $x^i = \left[x_1^i, \ x_2^i, \ \ldots, \ x_N^i\right]$ and velocities $v^i = \left[v_1^i, \ v_2^i, \ \ldots, \ v_N^i\right]$ (i=1,2,…,N) from uniform distributions with $x_i^j \in \{x_{\min}, x_{\max}\}$, and $v_i^j \in \{0, 0.1 \times (x_{\max} - x_{\min})\}$, j=1,2,…,N.

Step 2. Evaluate the fitness function values by $F_t(x^i)$ assigning each x^i as the neural network weights and the controller's parameters. Assign the global and local best positions: Set the local best position for each particle using $pbest^i = x^i$ and compare the evaluated fitness values and find the global best position $gbest^i = x^i$, for some $1 \le j \le N$, such that $F_t(x^j) \le F_t(x^i)$ for i=1, 2… N.

Step 3. Search for minimum value of F_t

i. Update the particle velocities v^i according to equation (11).

ii. Update all positions x^i using formula (13). Check all positions to ensure that $x_{\min} \le x_j^i \le x_{\max}$. Otherwise fly-back algorithm has to be implemented.

iii. Evaluate $F_t(x^i)$, (i=1,2,…,N).

iv. Update the local best position: if $F_t(x^i) < F_t(pbest^i)$, then $pbest^i = x^i$.

v. Update the global best position *gbest*, by letting $pbest^i = x^i$, for some $1 \le j \le N$ such that $F_t(x^j) < F_t(x^i)$, for (i=1,2,…,N).

Step 4. Repeat Step.4 until a goal is reached or the number of iterations is surpassed.

VI. RESULT AND DISCUSSION

In order to evaluate the effectiveness of the newly proposed scheme, a simulation example is considered. A simulation model of the coupled tank systems with its controllers was studied under Matlab/Simulink and the corresponding results are presented. The corresponding parameters of the nominal linear model were as follows: the cross sectional area of tank 1 and tank 2 were given as $A = 66.25\ (cm^2)$ with each height $H = 18.5\ (cm)$. The area of the coupling orifice was given as $a = 0.1963\ (cm^2)$. Meanwhile, the valve ratio at the outlet of tank 1 and 2 (β_1, β_2) were given as 0.35903 and 0.345848, respectively. The valve ration of the outlet between tank 1 and tank 2, β_x was set to 0.38705 with the gravitational rate, g =981cm/s^2.

The simulation was performed between [0:1500]s with a sampling interval taken as 1s. The liquid levels of the coupled tank system are required to follow step set-points within the range of 0 to 300 mm (0~100%). Process variables 1 and 2 (PV1 and PV2), namely the liquid level for both tank 1 and 2, respectively, were observed. In the training stage, the parameters of the PSO were initialized as follows. Population size = 20, inertia weight factor ω was set according to (12) where $\omega_{max} = 0.9$ and $\omega_{min} = 0.1$. Cognitive and social learning constants were $\Gamma_1 = \Gamma_2 = 1.4$. The time constants for the model reference were chosen as $\tau_1 = \tau_2 = 20$s. The maximum number of iterations was set as $iter_{max} = 200$.

In the first investigation, the performance of the controllers was measured by observing the PVs effect due to the changes in the set-point. The set-point in tank 2 was varied from its steady state condition of 120 cm with instantaneous step change by 50% up to 180 cm while the set-point in tank 1 remained constant. The PV responses show that hybrid PI-NN exhibits faster reaction to the step change as shown in Fig.7 and Fig.8. In addition, it can also be identified that hybrid PI-NN controller was very much fast and superior in achieving minimum steady state error as compared to other types of control schemes discussed. This is proven from the simulated results of squared errors (e^2) for both PV1 and PV2 as illustrated in Fig.9 and Fig.10.

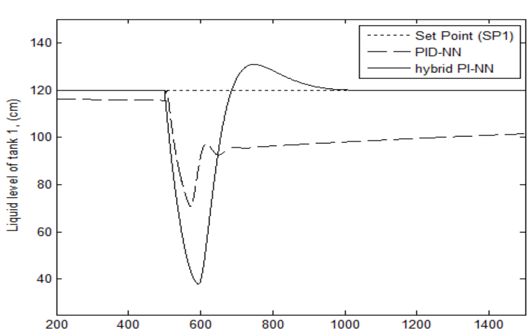

Figure 7: Simulated response of PV1 with step change in SP2

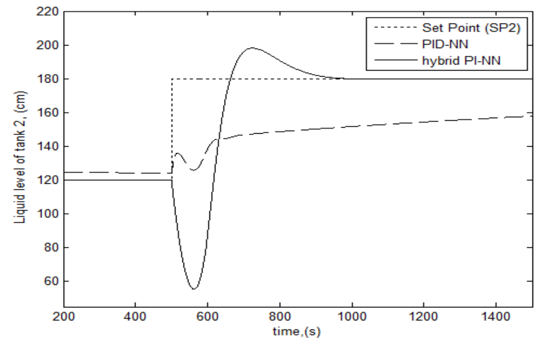

Figure 8: Simulated response of PV2 with step change in SP2

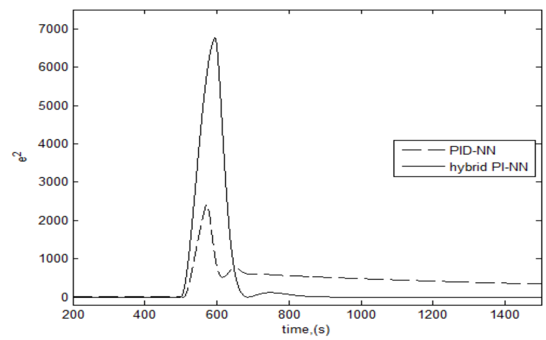

Figure 9: Squared error (e^2) change in PV1

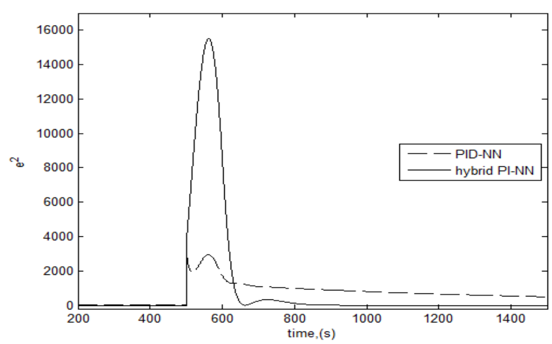

Figure 10: Squared error (e^2) change in PV2

The investigation was further carried out by introducing an impulse disturbance on channel PV1 for 5s at t=700s during steady-state. From the responses shown in Fig.11 and Fig.12, it can be summarized that PID-NN results better performance in rejecting the disturbance effect but comparatively equal in minimizing the system's errors. This is due to robustness of connective weights of the PID-NN which function to maintain the output responses for any external disturbance introduced. The performance comparison for hybrid PI-NN and PID-NN in terms of Integral Square Error (ISE) and Integral Time Absolute Error (ITAE) is shown in Fig.13. From the result, it can be summarized that both controllers are comparatively equal in terms of ISE but each surpass one another when comparing ITAE for level system in tank 1 and 2.

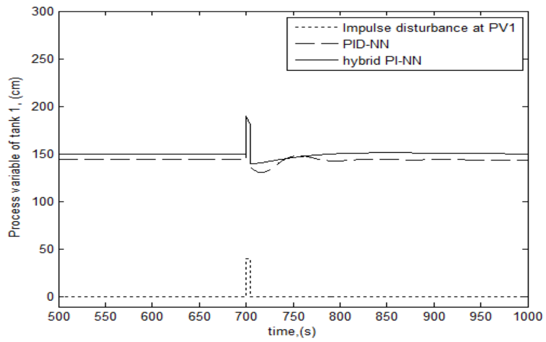

Figure 11: Simulated response of PV1 due to impulse disturbance in PV1

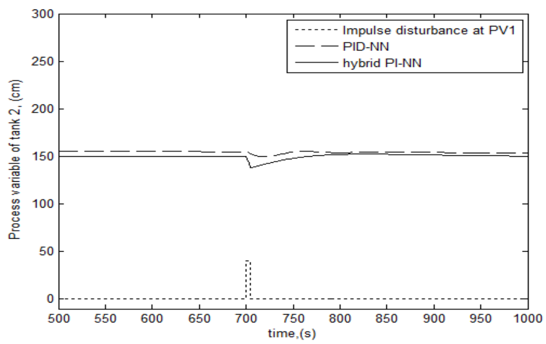

Figure 12: Simulated response of PV2 due to impulse disturbance in PV1

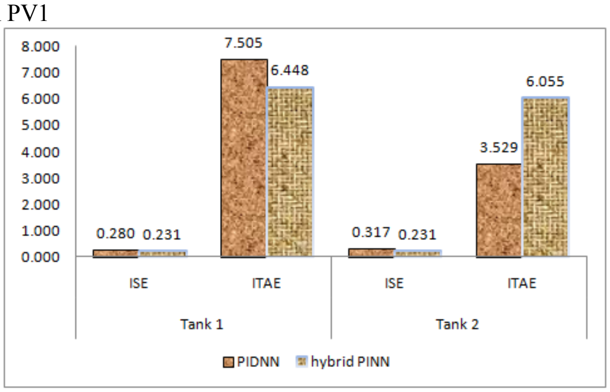

Figure 13: Performance comparison for proposed controllers based on ISE and ITAE (x10^6)

V. CONCLUSION

In this study, two neural network controllers, mainly hybrid PI-NN and PID-NN are introduced in controlling the liquid level TITO coupled tank system. The connective weights of the neural network structured are obtained by utilizing the Particles Swam Optimization through reference model adaptation. From verification via simulation results, it can be concluded that hybrid PI-NN is better in tracking the set-point change and achieving minimum steady-state error compared to PID-NN. However, if comparing in terms of disturbance rejection, it can be concluded that the PID-NN is superior to its counterpart.

ACKNOWLEDGMENT

This work was supported by Faculty of Electrical & Electronics Engineering, University Malaysia Pahang, under Control & Instrumentation (COINS) Research Group.

REFERENCES

[1] F.G. Shinskey, "Averaging level control," *Chemical Processing*, vol.60, no.9, Sep 1997, pp. 58-58.

[2] H. Pana, H. Wonga, V. Kapilaa, M.S. de Queiroz, "Experimental validation of a nonlinear backstepping liquid level controller for a state coupled two tank system", *Contro Engineering Practice*, vol. 13, no. 1, Jan. 2005, pp.27-40

[3] N. Guigues, C. Berho, J.-C. Foucher, A.-M. Fouillac, "The use of tank experiments in assessing the performance of 7 emerging tools in optimizing water monitoring," *Trends in Analytical Chemistry*, vol. 26, no. 4, April 2007, pp. 268-273.

[4] S.Aydin, S.Tokat,"Sliding mode control of a coupled Tank System with a state varying sliding surface parameter", International Workshop on Variable Structure System, June 2008,pp.355-60

[5] A.Numsomran, T.Suksri, M. Thumma, "Design of 2-DOF PI controller with decoupling for coupled-tank process", *International Conference on Control, Automation and Systems,* Seoul, Korea, Oct.2007, pp.339-344

[6] S.Tunyasrirut, T. Suksri, A. Numsomran, S Gulpanich, and K. Tirasesth, "The Auto-Tuning PID Controller for Interacting Water Level Process," *Proceedings of World Academy of Science, Engineering and Technology*, vol.12,March 2006, pp.134-138.

[7] H.Delavari, and A.R. Noiey, "Genetic-based Fuzzy Sliding Mode Control of an Interconnected Twin-Tanks," *The International Conference on Computer as a Tool (EUROCON, 2007)*, 9-12 Sep 2007, pp. 714-719.

[8] K.C. Tan, and Y. Li, "Performance-based control system design automation via evolutionary computing," *Engineering Applications of Artificial Intelligence*, vol.14, no.4, August 2001, pp. 473-486.

[9] R.C.Eberhart, and J. Kennedy,"A new optimizer employing particles swarm theory", *Proceedings of 6th International Symposium Micro Machine and Human Science*, Oct. 4-6, Nagoya, Japan. Piscataway,1995, pp: 39-43

[10] R.Muangsong, D.Koolpiruk, A. Khantachawana, P. Niranatlumpong,"A particle swarm optimization approach for optimal design of PID controller for position control using shape memory alloys", *Proceedings of ECTI-CON 2008*, pp.677-680

[11] H.W.Ge, Y.C.Liang, M.Marchese," A modified particle swarm optimization-based dynamic recurrent neural network for identifying and controlling nonlinear systems", *Journal of Computers and Structures*, Vol.85, 2007, pp.1611-1622

[12] R.W.Cui, J.Y.He,Q.H.Zhi, W.Z.Jiang, J.S.Chen," A modified particle swarm optimization algorithm and its application in optimal power flow problem", *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, August 2005, pp.2885-2889

[13] R.Poli, "Analysis of the Publications on the Applications of Particles Swarm Optimization", *Journal of Artificial Evolution and Applications,*2008,pp 1-10.