Online Self Tuning PID Control Using Neural Network for Tracking Control of a Pneumatic Cylinder Using Pulse Width Modulation Piloted Digital Valves

Sherif A. Elbelady\(^{a}\), H. E. Fawaz\(^{b}\), A. M. Abdul Aziz\(^{a}\)

\(^{a}\)Mech. Eng. Dept., Faculty of Engineering, Ain Shams University, Egypt.
\(^{b}\)Mech. Eng. Dept., National Research Centre, Dokki, Cairo, Egypt.

Email: shoshadel2005@gmail.com, ehf20012001@hotmail.com, abdelaziz_morgan@eng.asu.edu.eg.

Abstract—This study concerns experimental work for pneumatic position control using online self-tuning PID controller with a neural network (denoted as STNPID controller) using back propagation scheme. The actuator under control is a vertical double-acting single-rod pneumatic cylinder of 158 mm stroke and 20 mm bore diameter. Inexpensive high speed digital control valves (up to 150 Hz) are implemented to accurately control the cylinder piston position using Pulse Width Modulation (denoted as PWM) technique. The STNPID controller is a technique to apply neural networks for online tuning of the PID controller’s gains in a way of human tune the gains depending on the environmental and systems requirements in order to make the nonlinear system unaffected by the unpredictability of system’s parameters and disturbances such as noise. The inputs of STNPID include the sequences of tracking error, set point and control action, while the output of the neural network is optimized gains \(k_p\), \(k_i\), and \(k_d\) developed by Error back-propagation method. Results show that the STNPID controller is able to track both constant and variable set point trajectories efficiently by the pneumatic actuator system. By comparing the results of STNPID with the conventional PID controller showed that self-tuning of the PID gains can treat with the nonlinearity of the pneumatic system which is incompatible with a linear behavior of the conventional PID controller, so the tracking performance of the pneumatic cylinder enhanced with STNPID controller. For step response, the overshoot and steady state error decrease while the rising and descending times remain unchanged. The oscillation behavior for the sinusoidal response of the conventional PID controller is damped by The STNPID controller and better performance is observed. Also, the actuator sinusoidal wave response acts smoothly and uniformly compared to the conventional PID controller with more oscillatory actuator performance.

Index Term-- Pneumatic, digital valves, PID control, neural network, self-tuning and intelligent control.

1. INTRODUCTION

Position control of pneumatic systems is tricky to achieve precisely in many circumstances due to the inherited nonlinearity caused by air compressibility, piston dry friction and the nonlinear discontinuous regime of air flow through the control valves. However, it is used in many industrial applications due to its great advantages as it is reliable, high strength, clean, fast acting, self-cooling actuators, good power to weight ratio, compactness, ease of maintenance, readily available and cheap power source. Concentrated attention has been developed to the proper controller to tackle this problem that can deal with the nonlinearity of the pneumatic system. Tor Steinar Schei [1] developed the automatic tuning of PID controllers based on the estimation of a parametric transfer function model. The transfer function parameters are chosen to obtain an accurate estimation in a narrow frequency range. One of The auto-tuning method disadvantages was the inability to adapt the process continuous change. Chan and Leong [2] used four parameters to define the response characteristics; the normalized peak rise time, normalized overshoot, normalized peak to peak height and normalized final error. These parameters are taken as inputs for the learning process of the neural networks. This study investigated the use of different single, double and triple hidden layer back propagated neural networks. The best result was found at the layer’s structure of 4-10-10-8-2 for tuning of a representative process. Hamiti et al. [3] dealt with two parts of the controller, the first is analog P controller part (inner loop) and the second is a digital PI controller part (outer loop). The first controller used to make the system stable and the second used to specify the characteristics of the whole system. Qing-Guo Wang et al. [4] provided an auto-tuning method with feedback for multivariable PID controllers. The PID parameters of the controllers are determined individually. The proposed method is applied to various typical processes, where significant performance improvement over the existing tuning methods is demonstrated. Ming-Chang Shih and Ming-An Ma [5] provided a method of control for a pneumatic rod-less cylinder by using modified differential PWM method combined with fuzzy control technology. The performance of the controller is not affected by loading and gives satisfactory results. Han Koo Lee et al. [6] used a 5-port proportional valve for tracking position control of a pneumatic actuator. The proposed controller has an inner pressure control loop and an outer position control loop. The position controller was a PID controller using friction compensation with either neural network or nonlinear observer. The practical experiments showed that the controller has enhanced tracking performance and accuracy. Rachkov et al. [7] apply a Conventional and fuzzy logic controls for pneumatic manipulator problems in
positioning, both for angle trajectories and for long linear trajectories, used in construction tasks and carried out experimental testing. The obtained results allow widening the application range of pneumatic manipulators in construction. No-Cheol Park et al. [8] addressed a position control problem of a two-degree-of-freedom arm system having a flexible second link with artificial pneumatic muscle-type actuators. A composite controller design method is proposed in the framework of the singular perturbation method. Various robust control schemes are designed in order to meet with payload variation, parameter uncertainty, unmodeled vibration mode and actuator dynamics both in the slow and the fast subsystems. Wei-Der Chang et al. [9] used the auto-tuning method to select the multivariable PID controller gains according to the state by the steepest descent method using a simple structure neural network. This proposed method was used in several workable applications. TU Diep Cong Thanh and Kyoung Kwan Ahn [10] used the PAM (pneumatic artificial muscle) manipulator with a combination of conventional PID controller and the neural network to perform an accurate position control with the capability of learning, adaptation and disturbance rejection. This research achieved online control with a simple structure and better dynamic property. The results showed that the nonlinear PID controller using neural network was efficient way to evolve human-robot by using the PAM manipulator. Arcangelo Messina et al. [11] studied the control of pneumatic actuator using PWM technique and on-off solenoid valves. The authors made a mathematical model also. The comparison between the analytical and experimental results showed that the ability of the theoretical model to give an accurate mean position of the actuator less than about 2 mm. Murad Samhouri et al. [12] used an online tuning to make a relationship between the controller gains and the target output response by using an adaptive neuro-fuzzy inference system (ANFIS). Kyongkwon Ahn and Shinichi Yokota [13] used a modified pulse width modulation [MPWM] technique with smooth switching control algorithm to enhance the controller performance when making an abrupt change of external loads. The modified PWM technique allows replacing the costly servo valves with on/off solenoid valves, increases the control performance, compensates the dead-time of on/off valves and reduces the steady state errors within 0.2 mm. Linear vector quantization neural network [LVQNN] used as switching algorithm for control parameters to classify the condition of external loads and to select the controller gains. The author used position, velocity, and acceleration feedback to enhance controller performance. Wei-Der Chang [14] used a modified formula in genetic algorithms (GAs) to determine the PID controller gains to control process variables. The genes represent gains and then form a chromosome. He concluded that GA gives better output response and minimum IAE (Integrated Absolute Error) values. Giulio D’Emilia et al. [15] results concluded that the auto-tuning of PID controller based on a neural network approach reduced the time duration necessary for auto-tuning in order of magnitude with respect to traditional methods. Mohammed Hassan and Ganesh Kothapalli [16] made a comparison between Neural Network based PI controller and Neural Network based PID controller using a pneumatic servo actuator. The feedback is the changes in position error and changes in external load force. The results were in Favor of Neural Network based PID controller. The neural network based PID controller has a simply trained structure with less MSE (Mean squared error) than PI type. Xie and Jamwal [17] used a Pneumatic Muscle Actuators (PMA) controlled by a feedforward fuzzy controller with Genetic Algorithm (GA) parameters setting. The proposed iterative controller was able to control the PMA with high accuracy and excellent trajectory tracking performance at different desired trajectory. Thananchai Leephakpreeda [18] Made a Pneumatic Artificial Muscle (PAM) model to study the dynamics and earn controller design knowledge. The PAM actuators position and force manipulated by the Proportional-Type Fuzzy Control law based on a minimum time control design and used to determine the mass flow rate of compressed air. The estimation of inaccessible variables acquired from a neural

<table>
<thead>
<tr>
<th>NOMENCLATURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>e</td>
</tr>
<tr>
<td>t</td>
</tr>
<tr>
<td>$k_p$</td>
</tr>
<tr>
<td>$k_i$</td>
</tr>
<tr>
<td>$k_d$</td>
</tr>
<tr>
<td>$T_i$</td>
</tr>
<tr>
<td>$T_d$</td>
</tr>
<tr>
<td>$u(t)$</td>
</tr>
<tr>
<td>$y_d$</td>
</tr>
<tr>
<td>$y_a$</td>
</tr>
<tr>
<td>$E$</td>
</tr>
<tr>
<td>$O_k$</td>
</tr>
<tr>
<td>$\theta_j$</td>
</tr>
<tr>
<td>$O(x)$</td>
</tr>
<tr>
<td>$\Delta w$</td>
</tr>
<tr>
<td>$\delta$</td>
</tr>
<tr>
<td>$\text{net}_x$</td>
</tr>
<tr>
<td>$O_i$</td>
</tr>
<tr>
<td>$O_j$</td>
</tr>
<tr>
<td>$w_{kj}$</td>
</tr>
<tr>
<td>$w_{ji}$</td>
</tr>
<tr>
<td>$\eta$</td>
</tr>
<tr>
<td>$\alpha$</td>
</tr>
<tr>
<td>$E_{\text{RMS}}$</td>
</tr>
<tr>
<td>$N$</td>
</tr>
<tr>
<td>$\theta_k$</td>
</tr>
</tbody>
</table>
network instead of direct measurement. Mohannad Farag and Norsinnira Zainul Azlan [19] developed a cascade control system by combining a conventional PID control with the adaptive backstepping position control for a pneumatic anthropomorphic robotic hand. The present paper deals with the experimental position control of a pneumatic cylinder using The STNPID controller with PWM technique using error back propagation algorithm. This PWM technique allows using on/off solenoid valves instead of servo valves so that the gains are updated continuously in order to enhance the tracking performance of the pneumatic cylinder.

2. Design of STNPID Controller

2.1 PID controller

The transfer function of the most basic form of PID controller is:

\[ u(t) = k_p \left[ e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de(t)}{dt} \right] \]  

(1)

\[ k_i = \frac{k_p}{T_i} \]  

(2)

\[ k_d = k_p T_d \]  

(3)

The error \( e(t) \) is defined as:

\[ e(t) = y_d(t) - y_a(t) \]  

(4)

The aforementioned PID equations were modeled in labview software and do the tuning process manually to get the best performance of the PID controller.

2.2 STNPID controller

The neural networks are capable of generalizing and learning dynamic relationships between the inputs and outputs of the plant. Furthermore, the neural networks can constantly update their connection weights to respond to changes in the plant dynamics. The self-recursive learning method feature of the neural networks can be exploited in auto-tuning the PID gains where there are nonlinearities that cannot be expressed in closed form or some lost dynamic modes. The aim of the STNPID controller design is to provide good tracking performance for the position of the pneumatic actuator. Fig.1 shows the STNPID controller structure which is built by combining PID and neural network with error back propagation algorithm. The neural network inputs are an error, set point and control action. The neural network performs online self-tuning using error back propagation technique to develop PID gains. Error backpropagation is a technique of neural networks training looks for the minimum weight space error function of the network using the gradient descent method. In the controller, the three-layered neural network is built as shown in Fig.2 three neurons in input layer are chosen corresponds to error, set point and control action. Four neurons in the hidden layer and three neurons in the output layer correspond to PID gain parameters. We assume a three-layered neural network whose input is \( r_i(t) \) and the output at the output layer is \( O_k(t) \).

\[ r_i(t) = \begin{bmatrix} \epsilon \\ y_d \\ u \end{bmatrix} \]  

(5)

\[ O_k(t) = \begin{bmatrix} k_p \\ k_i \\ k_d \end{bmatrix} \]  

(6)

\[ E = \frac{1}{2} e^2 (t + 1) = \frac{1}{2} [y_d(t + 1) - y_a(t + 1)]^2 \]  

(7)

\[ \Delta w_{kj}(t + 1) = -\eta \frac{\partial E}{\partial w_{kj}} + \alpha \Delta w_{kj}(t) \]  

(8)

\[ \Delta w_{ij}(t + 1) = \eta \delta_i O_j + \alpha \Delta w_{ij}(t) \]  

(9)

\[ \Delta w_{ij}(t + 1) = -\eta \frac{\partial E}{\partial w_{ij}} + \alpha \Delta w_{ij}(t) \]  

(10)

The second term \( \alpha \Delta w_{kj}(t) \) called Relaxation Term.

\[ \delta_k = -\frac{\partial E}{\partial \text{net}_k} \]  

(12)

\[ \text{net}_k = \Sigma w_{kj} O_j + \theta_k \]  

(13)

\[ \delta_i = -\frac{\partial E}{\partial \text{net}_i} \]  

(14)

\[ \text{net}_i = \Sigma w_{ji} O_j + \theta_i \]  

(15)

\[ O(x) = f(\text{net}_x) = \frac{1}{1 + e^{-\text{net}_x}} \]  

(16)

To minimize the error:

\[ \frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial \text{net}_k} \frac{\partial \text{net}_k}{\partial w_{kj}} \]  

(17)

\[ \frac{\partial E}{\partial \text{net}_k} \frac{\partial \text{net}_k}{\partial w_{kj}} \]  

(18)

\[ O_j = \frac{\partial \text{net}_k}{\partial w_{kj}} \]  

(19)

\[ \delta_k = -\frac{\partial E}{\partial \text{net}_k} = -\frac{\partial E}{\partial y(t+1)} \frac{\partial y(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial O(k)} \frac{\partial O(k)}{\partial \text{net}_k} \]  

(20)

\[ \frac{\partial O(k)}{\partial \text{net}_k} = f'(\text{net}_k) = \frac{O(k)(1 - O(k))}{O(t)} \]  

(21)

\[ \delta_k = \epsilon(t + 1) \frac{\partial y(t+1)}{\partial u(t)} \frac{\partial u(t)}{\partial O(k)} \frac{\partial O(k)}{\partial \text{net}_k} \]  

(22)

For the hidden layer, we have:

\[ \frac{\partial E}{\partial \text{net}_j} = \sum_k \frac{\partial E}{\partial \text{net}_k} \frac{\partial O_j}{\partial \text{net}_k} = -\sum_k \delta_k w_{kj} f'(\text{net}_j) = -\sum_k \delta_k w_{kj} O_j (1 - O_j) \]  

(23)

Thus, we have:

\[ \delta_j = -\frac{\partial E}{\partial \text{net}_j} = \sum_k \delta_k w_{kj} O_j (1 - O_j) \]  

(24)

The system jacobian \( \frac{\partial y(t+1)}{\partial u(t)} \) is required to calculate \( \delta_k \).
Fig. 1. STN PID controller structure.

Fig. 2. Three layers of the neural networks.
Where, $O(1) = K_p$, $O(2) = K_i$, $O(3) = K_d$

The STN PID controller algorithm can be summarized by the following steps:

1. Set $w_{kj}$, $w_{ij}$, $\theta_k$, $\theta_j$, $\eta$, and $\alpha$ initial values. Start with $t=0$.

2. Compute: $e(t+1)$ From eqn. (4)
   
   $\delta_k$ at $k = 1, 2, 3$ From eqn. (22)

   Where $\frac{\partial u(t)}{\partial x(k)}$ From eqn. (25)

3. Change the connection weight by Compute:
   $\Delta w_{kj}(t + 1)$ From eqn. (9)

4. Compute $\delta_f = \sum_k \delta w_{kj} O_j (1 - O_j)$.

5. Change the connection weight by Compute:
   $\Delta w_{lj}(t + 1)$ From eqn. (11)

6. $t \rightarrow t+1$ then go to step 2

The STN PID controller algorithm can be summarized by the following steps:

1. Set $w_{kj}$, $w_{ij}$, $\theta_k$, $\theta_j$, $\eta$, and $\alpha$ initial values. Start with $t=0$.

2. Compute: $e(t+1)$ From eqn. (4)

   $\delta_k$ at $k = 1, 2, 3$ From eqn. (22)

   Where $\frac{\partial u(t)}{\partial x(k)}$ From eqn. (25)

3. Change the connection weight by Compute:

   $\Delta w_{kj}(t + 1)$ From eqn. (9)

4. Compute $\delta_f = \sum_k \delta w_{kj} O_j (1 - O_j)$.

5. Change the connection weight by Compute:

   $\Delta w_{lj}(t + 1)$ From eqn. (11)

6. $t \rightarrow t+1$ then go to step 2

The STN PID controller algorithm can be summarized by the following steps:

1. Set $w_{kj}$, $w_{ij}$, $\theta_k$, $\theta_j$, $\eta$, and $\alpha$ initial values. Start with $t=0$.

2. Compute: $e(t+1)$ From eqn. (4)

   $\delta_k$ at $k = 1, 2, 3$ From eqn. (22)

   Where $\frac{\partial u(t)}{\partial x(k)}$ From eqn. (25)

3. Change the connection weight by Compute:

   $\Delta w_{kj}(t + 1)$ From eqn. (9)

4. Compute $\delta_f = \sum_k \delta w_{kj} O_j (1 - O_j)$.

5. Change the connection weight by Compute:

   $\Delta w_{lj}(t + 1)$ From eqn. (11)

6. $t \rightarrow t+1$ then go to step 2

3. EXPERIMENTAL SETUP

   A photograph of the experimental setup is shown in Fig. 3 while the corresponding schematic diagram is shown in Fig. 4. The experimental test rig is mainly composed of a double-acting single-rod pneumatic cylinder of 158 mm stroke and 20 mm bore mounted in vertical position with no load. By experimental measurements, we found that the static friction force for the cylinder is 29.43 N, the kinetic friction force is 17.66 N and the accumulator volume is 10.996 × 10^{-5} m^3. The supply air pressure was set at 6 bar gauge pressure by using the pressure regulator. Two groups of digital valves (IV1&IV2) operated at a high frequency up to 100 Hz used to supply air to the cylinder in upward and downward directions. Each group consists of two identical valves operated with PWM technique. Another two digital valves (OV1 &OV2) which operated at either fully opened or fully closed mode are used to release air from the cylinder. Piston position is measured by a linear potentiometer with a maximum linearity of 0.5%. Labview application is used as computer software to control the digital valves and exchange input and output signals through National Instrument interface card (NI-USB609). The air digital valve, Fig. 5, has a bandwidth from 0 to 100 Hz with a maximum lift of 0.1 mm. Power amplifier circuit and PWM circuit lead the valves for an analysis of the consequences of different stages of PWM technique for control requirements as shown in Fig. 6. The PWM was produced as follows:

1. A clock of a frequency of 50-300 Hz leads an 8-bit binary counter.

2. A digital to analog converter executes the summation of the 8-bitary levels to manufacture a saw tooth signal.

3. A comparator contrasts the saw tooth signal and the control action voltage levels to produce the final variable duty driving signal.

4. The valve frequency, 95 Hz, was chosen to operate within the allowable bandwidth to guarantee the correct valve operations.

5. The power circuit consists of three stage transistors feed the solenoids with the desired power for their operations.
Fig. 4. Schematic diagram of the experimental setup
Fig. 5. Digital valve construction

Fig. 6. Analysis of the consequences of different stages of PWM technique
The aforementioned equations are modeled using the Labview software which consists of three modules:

The first module is concerned with interfacing the LabVIEW program with NI-DAQ. Five interfacing modules are used during the experiments as follows:

1) One analog input is connected to the linear potentiometer to measure the piston displacement.
2) Two analog outputs are used to drive the inlet valves (IV1 & IV2) via PWM and amplifier circuit.
3) Two digital outputs are used to drive the valves (OV1 & OV2) which have two positions only. (fully opened or fully closed)

The second module is the STNPID controller module which was discussed in detail in section 2.2.

The third module creates the reference wave signals which take two input shapes; step and sinusoidal waves.

4. Results

To initiate the training process, arbitrary values for the PID gain parameters could assign and used as initial values. Then the PID parameters are modified and adjusted through online learning technique of neural network algorithm. The results depicted in Fig. 7 shows a comparison between the performance of step up response to the conventional PID and the STNPID controller at $K_p$=2.4, $K_i$=0.2, and $K_d$=0.02. From Fig. 7, it is clear that the system with conventional PID has to be stable otherwise overshoot & steady state error is high as it is 9.6 mm and 2.3 mm respectively. The STNPID controller was designed to make the system more stable from the hands of overshot and steady state error so it became too small, as it was 8 mm and 0.7 mm respectively. The rise time of the system is nearly the same for both the conventional PID and the STNPID controller. So, self-tuning of PID gains for pneumatic tracking control is useful in minimizing the overshoot and steady state error but has no effect in rising time. Figs. 8 and 9 show the actuator control action corresponding to the step input in terms of inlet valve1 response shown in Fig 8 and inlet valve 2 responses shown in Fig. 9. Fig. 8 shows that the inlet valve1 response goes from the fully closed to the fully opened position before it stabilizes. This suggests to significant changes in the set points imposed a large error produced by the controller, thus causing a rapid actuator movements over a short period of time. Otherwise, the tracking operation of the controller is stable and tracks the desired set point. The inlet valve 2 is fully closed during the step response test except for the overshoot period where the inlet valve 2 acts to drive the piston to the opposite direction in order to damp the overshoot. Fig. 10 shows a comparison between the performance of step down response for the conventional PID and the STNPID controller at $K_p$=2,$K_i$=1, $K_d$=0.05. For conventional PID controller the steady state error =1.24mm, the overshoot is 2mm and nonuniform tracking performance is observed. While for the STNPID controller the steady state error=0.6 mm with no overshoot and uniform tracking performance. Figs. 11&12 shows the actuator control action for step down response in terms of inlet valve1 response as shown in Fig.11 and inlet valve2 response as shown in Fig.12. Inlet valve1 is fully closed during the operation except for overshoot period. The inlet valve 2 has a faster response from a fully closed to fully open position. Which cause actuators rapid movement over a short period of time. The nonuniformity response of the inlet valve2 is observed, which reflects the inability of the conventional PID controller to deal with the non-linearity of the pneumatic systems, while the smooth operation is observed in the STNPID controller. Fig 13 shows a comparison between the conventional PID and the STNPID controller tracking performance for a sinusoidal wave set point at a frequency of 0.1 Hz and PID gains at $K_p$=6, $K_i$=3, $K_d$=0.06. The magnitude of oscillations in the conventional PID controller is somewhat large due to the fact that the pneumatic system is nonlinear which is not matched with the linear behavior of the conventional PID controller while the self-tuning criteria of the STNPID controller lead to better performance and reduce the oscillations. For this reason, the actuator responses for conventional PID controller seems to be more oscillatory as shown in Figs. 14 and 15 for inlet valve 1 and inlet valve 2 respectively. While these oscillations are damped and the actuator response acts smoothly for The STNPID controller, which indicates that the self-tuning criteria enhance the actuator performance. As shown by the experimental results, the STNPID controller provides a sophisticated approach with a smooth and fast response, without any large overshooting compared with conventional PID controller. Similar trends and conclusions are observed at $K_p$=7, $K_i$=1, $K_d$=0.05 for the comparison between the conventional and proposed controllers with the sinusoidal wave set point at a frequency of 0.1 Hz as shown in Fig. 16 and the corresponding actuator control action are displayed in Figs. 17&18 for inlet valve1 and inlet valve2 respectively.

The amount of the Root Mean Square Error for both conventional PID and STNPID controllers at different PID gains are tabulated in Table1 and 2 for Square and Sinusoidal wave respectively using the following formula:

$$E_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} e_i^2}$$

(26)

Tables 1 and 2 show that self-tuning of PID gains via STNPID controller has reduced the amount of root mean square errors at the expense of the conventional PID controller at both Square and Sinusoidal waves for all values of PID gains, which indicates that the self-tuning method of the PID gains via STNPID controller enhances the tracking performance of the pneumatic actuator at any reference waveform and PID gains.
Fig. 7. Step response at $K_p=2.4$, $K_i=0.2$, $K_d=0.02$

Fig. 8. Inlet valve 1 response at $K_p=2.4$, $K_i=0.2$, $K_d=0.02$

Fig. 9. Inlet valve 2 response at $K_p=2.4$, $K_i=0.2$, $K_d=0.02$
Fig. 10. Step response at $K_p=2$, $K_i=1$, $K_d=0.05$

Fig. 11. Inlet valve 1 control action at $K_p=2$, $K_i=1$, $K_d=0.05$

Fig. 12. Inlet valve 2 control action at $K_p=2$, $K_i=1$, $K_d=0.05$
Fig. 13. Response for 0.1 Hz sinusoidal wave set point at $K_p=6$, $K_i=3$, $K_d=0.06$

Fig. 14. Inlet valve 1 control action for 0.1 Hz sinusoidal wave at $K_p=6$, $K_i=3$, $K_d=0.06$

Fig. 15. Inlet valve 2 control action for 0.1 Hz sinusoidal wave at $K_p=6$, $K_i=3$, $K_d=0.06$
Fig. 16. Response for 0.1 Hz sinusoidal wave set point at $K_p=7$, $K_i=1$, $K_d=0.05$

Fig. 17. Inlet valve 1 control action for 0.1 Hz sinusoidal wave at $K_p=7$, $K_i=1$, $K_d=0.05$

Fig. 18. Inlet valve 2 control action for 0.1 Hz sinusoidal wave at $K_p=7$, $K_i=1$, $K_d=0.05$
5. CONCLUSION

A new adaptive tuning method for PID gains controller based on corresponding online auto-tuning neurons has been proposed. So that the gains are updated continuously. Experimental position control of a pneumatic cylinder is achieved using STNPID controller. The experimental results showed that the performance of STNPID controller can be summarized as:

1- For step response the STNPID controller minimizing the overshoot and steady state error while the rising and descending times remain unchanged.

2- The magnitude of oscillations of the sinusoidal wave response in the conventional PID controller is large due to the inability of the controller to treat the nonlinearity of the pneumatic system while the self-tuning criteria of the STNPID controller can damp the oscillation and produce better performance.

3- The actuator response for the sinusoidal wave acts smoothly and uniformly compared to the conventional PID controller with more oscillatory actuator performance.

4- The STNPID controller enhances the tracking performance of the pneumatic actuators over the conventional PID controller at any reference waveform and PID gains.

REFERENCES