An Improved Single Neuron Adaptive PID Controller System Based on Additional Error of an Inversed-Control Signal

Benyamin Kusumoputro and Muhammad Rif'an
Computational Intelligence and Intelligent System, Department of Electrical Engineering, Universitas Indonesia, Depok 16424, Indonesia

This paper presents a novel single neuron adaptive PID controller system that developed by utilizing an additional error of the inversed-control signal from the plant output on its learning mechanism, in order to update the neuron connection weights. This additional error is calculated from the difference between an inversed-control signal through an inverse neural network and the input control signal of the plant. Actually, this additional error is derived mathematically through a new quadratic error performance function that is utilized for learning the single neuron adaptive PID controller. By using this new quadratic error performance function of the controller system, the convergence speed of the adaptive neuron is increased and the control performance of the novel single neuron adaptive PID controller system is greatly improved. The control system parameters such as the rise time, settling time, and the overshoot of the system of the proposed algorithm have been compared with that of the classical single neuron adaptive PID controller system; and the simulation results show that the control effect of this novel controller system outperformed the classical single neuron adaptive PID controller, especially in its strong robustness and better self-adaptation characteristics.

Keywords: Single Neuron, Adaptive Control, PID Control, Inverse Neural Network.

1. INTRODUCTION
The classical PID control is one of the most utilized techniques in the standard process control due to its simple structure and high reliability. This PID controller, however, could not easily adapt to the problem of time-variation and the non-linearity of the industry processes, because the parameters has been already fixed with a derived constant value.

In recent years, artificial neural network (ANN) has been reported to be used as a control system for a time-dependent nonlinear system. The ANN-based controller system has been usually applied for state feedback controller design, nonlinear system control, nonlinear dynamical system identification, and optimal control synthesis. The ANN is a massive parallel distributed processor made up of a simple processing neuron for memorizing the knowledge and making it available after training the networks weights through a determined learning algorithm. A simple but powerful neural network is a multi-layer perceptron (MLP) with one hidden layer, trained by using a back-propagation learning mechanism for updating the artificial neural networks parameters. However, the learning mechanism of the ANN needs higher computational cost, which hampering the used of this technique as a real-time control system.

In order to improve the performances of ANN-based control system, especially in reducing the networks learning time, a single neuron adaptive PID control, which is the hybrid of neuron learning mechanism with a classical PID control technique, has been proposed. Beside the structure of a single neuron adaptive PID controller is very simple, the controller can learn and be adjusted on line during the working process, and hence it can be utilized in a complex environment variation of a real plant.

Various neural networks weight update algorithm are proposed for a single neuron adaptive PID (SNA-PID) controller system, such as Hebb learning rule, quadratic object function, and auto gain regulation. However, even though the SNA-PID controller system has a self-adaptation and robustness capability, the system performance parameters, such as rise time, settling time, and overshoot, is still open for improvement.

As the system performance parameters are depend on the defined quadratic parameter function to be minimized, in this paper, a new quadratic performance function to be minimized is proposed. This new quadratic performance function is accomplished by using an additional error function that calculated from the difference between the input control signal and the actual...
control signal as an inverse form from the actual output of the plant. As the consequence of the proposed new quadratic performance function, the architectural structure of the single neuron adaptive PID is changed accordingly, by an additional neural network as an inverse of the plant, which receive input from the plant output.

This paper is organized as follows. Section 2 presents the brief introduction of classical SNA-PID controller system, and Section 3 comprehensively discusses the method of the improved single neuron adaptive PID (ISNA-PID) controller system based on the novel quadratic parameter function. In section 4, both the simulation of the characteristics performance of the classical SNA-PID and the ISNA-PID controller systems are presented, followed by conclusions in Section 5.

2. CLASSICAL SINGLE NEURON ADAPTIVE PID CONTROLLER

Block diagram of the classical SNA-PID controller system is shown in Figure 1. The inputs of the state convertor are the difference between the reference input \( r(k) \) and the actual output \( y(k) \) of the plant. While the outputs of state convertor can be written as follows:

\[
\begin{align*}
x_1(k) &= e(k) - e(k-1) \\
x_2(k) &= e(k) \\
x_3(k) &= e(k) - 2e(k-1) + e(k-2)
\end{align*}
\]  

(1)

with the error at time \( k \) is \( e(k) = r(k) - y(k) \), the error at time \( k-1 \) is \( e(k-1) \), and for \( k-2 \) is \( e(k-2) \), respectively.

The outputs of the state convertor are then inputted to the single neuron acting like a PID, with the weights matrix of the neuron is defined as \( W = \begin{bmatrix} w_1, w_2, w_3 \end{bmatrix} \). The output of the neuron, with the gain \( K \) can be written as:

\[
\Delta u(k) = K(w_1x_1(k) + w_2x_2(k) + w_3x_3(k))
\]  

(2)

The SNA-PID controller system has the same structure with that of a classical incremental PID control methods, however, as can be clearly seen from (2), the neuron weights can be adaptively adjusted online according to the defined performance function. Supposed the quadratic performance function to be minimized is defined as:

\[
P(k) = \frac{1}{2} [r(k + 1) - y(k + 1)]^2 = \frac{1}{2} e^2(k)
\]  

(3)

Using the gradient descent algorithm and the chain rule, the single neuron connection weights are updated by:

\[
\Delta w_i(k) = w_i(k + 1) - w_i(k) = -\alpha_i \frac{\partial P(k)}{\partial w_i(k)} = -\alpha_i K e_i(k) \frac{\partial y(k + 1)}{\partial u(k)} x_i(k), \quad i=1,2,3
\]  

(4)

where \( \alpha_i \) the learning rate for each weight updating. As for the unknown system, the term \( \partial y(k + 1)/\partial u(k) \) is unknown, thus a function \( \text{sgn}(\partial y(k + 1)/\partial u(k)) \) can be used instead.\(^{20}\)

The control signal is then calculated through:

\[
u(k) = u(k - 1) + \Delta u(k)
\]  

(5)

with

\[
\Delta u(k) = K \sum_{i=1}^{3} w_i(k) x_i(k)
\]  

(6)

3. AN IMPROVED SINGLE NEURON ADAPTIVE PID CONTROLLER SYSTEM

In the classical SNA-PID controller system, there are some problems that have to be encountered to optimize its function. First, the learning algorithm of the neuron is derived based on the gradient method, which is a typical one-order optimization method that shows a slow learning speed and easily to be trapped in a local minimum.

This disadvantage influenced the abilities of the algorithm to be used in fast tracking and anti-interference applications. Second, as there are three learning rate coefficients have to be adjusted, and this coefficient makes great influence to the performance of the neural network weight training, tuning those parameters is another limitation condition. Third, the value of coefficient \( K \) plays a vital role on the dynamic response and the stability of the system, so that tuning this coefficient increased the difficulty of using this method.

As the coefficient value of \( K \) is bigger, the dynamic response time will be shorter, which makes the overshoot is increasing, and control time is longer. When the coefficient value of \( K \) is smaller, however, the overshoot will decrease accordingly, but the system response slows down. Moreover, if the coefficient value of \( K \) is too small, the response of the system could not properly track the reference signal.\(^{21}\)

As tuning the optimize coefficient of those parameters could not be conducted for all of the working range of the plant, the expected output control signal from this controller system may not quite well defined. As the consequence, the actual output of the plant may differ from the reference input signal, producing an error that difficult to be adaptively adjusted by the controller system.

To overcome the drawbacks of the classical SNA-PID controller system, an improved novel architectural structure of a single neuron adaptive PID controller system as can be seen in Figure 2 is proposed. In this novel controller system, the quadratic performance function that used to update the neuron connection weights is changed by an additional error function generated by the difference between the input control signal of the plant and the inverse-control signal from the actual output from the plant.
Detail of the quadratic performance function improvement can be written as follow. In the classical SNA-PID system, the quadratic performance function to be minimized is written in (3), which just depend on the error derived through the difference between the reference input and the actual output of the system. When a disturbance occurred in the plant, the output signal is also changing accordingly, which could not be adjusted properly by using (3) of the classical SNA-PID system, due to a time lag of the input of state converter. In order to improve the time lag problem on the SNA-PID system, an inverse neural networks is used in our proposed ISNA-PID system, to generate an inverted-control signal directly from the output signal. And by calculating the difference between the inverses-control signal with the actual control signal this error control signal is then superposed to (3) to be a new quadratic performance function (8).

As can be clearly seen in Figure 2, the additional error function of $\Delta u$ is then inputted back into (3), where $\hat{u}$ is the output of inverse neural networks and $\Delta u(k) = \hat{u}(k) - u(k-1)$, yields a new quadratic performance function to be minimized as:

$$P(k) = \frac{1}{2} [(y(k+d) - \hat{u}(k))^2 + (\Delta u(k+1) - \Delta u(k+d))^2]$$

(8)

where $y(k+d)$ is the reference input system at the time of $k+d$, $\hat{u}(k)$ is the actual output of the system at the time of $k+d$, $\Delta u(k+d)$ is the output of the single neuron PID at the time of $k+d$, $\Delta u(k+1)$ is the output of the inverse neural network at the time of $k+1$, and $d$ is the time delay of system.

Based on the new quadratic performance function defined in (8), the weights of the neuron such as in (4) are now updated by:

$$w_1(k+1) = w_1(k) + \alpha_1 \frac{E(k)}{\partial x_1(k)} [\Delta u(k) - \Delta u(k+1)]$$

(9)

$$w_2(k+1) = w_2(k) + \alpha_2 \frac{E(k)}{\partial x_2(k)} [\Delta u(k) - \Delta u(k+1)]$$

(10)

$$w_3(k+1) = w_3(k) + \alpha_3 \frac{E(k)}{\partial x_3(k)} [\Delta u(k) - \Delta u(k+1)]$$

(11)

To provide an additional error function of $\Delta u - \hat{u}$ into the quadratic performance function, an inverse model of the plant is put in the architectural structure of the controller system. Inverse model is constructing using a neural network that represents the inverse of the system dynamics after a completion of training.

Figure 3 shows the utilization of ANN as an inverse model that consists of one input layer with 7 neurons, one hidden layer with 5 neurons, and one output layer one neuron, respectively. The output of the inverse model is given as:

$$\hat{u}(k) = f^{-1}(y(k+1), y(k), y(k-1), u(k), u(k-1))$$

(12)

where $\hat{u}(k)$ is the prediction of the control input, and $f^{-1}$ represents the inverse map of the forward model.

The input signals used to train and test the network consists of pseudo random multistep signals in the range 0 to 1, which is sampled at every 1 dimensionless time interval. The input variable, $u(k)$, and output variable, $y(k)$, are also scaled to be in the range between 0 and 1.

Back-propagation learning method that was introduced by Rumelhart21 is implemented for the training the neural network. The purpose of this back-propagation algorithm is to minimize a defined loss function based on the network output error. Minimization of the defined loss function is conducted by updating the network parameters in the negative gradient direction of the loss function. The recursive updates of the network parameters have the following form:

$$v_{k+1} = v_k - \alpha \frac{\partial E}{\partial v_k}$$

(13)

where $E$ is the defined loss function or the performance index to be minimized, $\alpha$ is the learning rate, with $\alpha > 0$ and $v$ is the network parameters or the connection weights between neuron, to be updated to minimize $E$.

The performance index to be minimized is given by,

$$E = \sum (u(k) - \hat{u}(k))^2$$

(14)

where

$$\hat{u}(k) = \sigma(v_k^T x_k)$$

(15)

And the partial derivatives of $E$ is given by,

$$\frac{\partial E}{\partial v_k} = \sigma'(v_k^T x_k) x_k$$

(16)

4. EXPERIMENTAL RESULT

In order to verify the performance of the improved single neuron PID controller and to compare with that of the classical single neuron PID controller, a simulation example using a MATLAB® is conducted. The Armature-controlled dc servomotor system presented by Ogata22 with a minor modification to be a nonlinear
system is used in this paper. The discrete-time nonlinear system plant is mathematically represented as:

$$y(k + 1) = 0.368y(k)^2 + 0.264y(k - 1) + 0.632u(k) \quad (17)$$

To define the inverse model that emulate the $f^{-1}(y, u)$, a total of 500 random data is generated for training the inverse neural network using back-propagation supervised learning. The learning rate of the back-propagation neural network is set to 0.01 and the momentum is 0.3, respectively. Figure 4 shows the testing result of the inverse model. It is clearly seen from this figure that the estimated value of the neural network is in good agreement with that of the actual value, with the mean-sum-square-error (MSSE) is 1.6826 $\times 10^{-4}$.

In the next experiment, for comparing the output characteristics of the ISNA-PID controller with that of the SNA-PID system, two kind of reference input signals, i.e., sinusoidal and step-function signals, respectively, are applied to the discrete-time nonlinear system plant. In the first experiment, a sinusoidal signal as the reference input is fed to the plant, with the initial weights of the single neuron PID controller are set to $w_1(0) = w_2(0) = w_3(0) = 0.5$, and $\alpha_1 = \alpha_2 = \alpha_3 = 0.3$, respectively, with $K$ is defined as 1.

The number of the simulation data is 600. Result comparison of the ISNA-PID and the SNA-PID is depicted in Figure 5. As can be clearly seen from this figure, both of the single neuron adaptive PID controllers are able to track the reference input with good agreement. However, as shown in the upscale picture, the ISNA-PID shows a faster response compare with that of the SNA-PID. This result shows that the improved single neuron adaptive PID outperformed the transient behavior of model system over the classical single neuron adaptive PID system.

In the second experiment, a step-function signal $r(k) = 1$ as the reference input is fed to the plant, with the initial weights of the single neuron PID controller are set to be the same with that of the first experiment. In this experiment, the simulation data is set to be 300, and the experimental result is depicted in Figure 6. As can be clearly seen from this figure, compared with that of the SNA-PID system, the dynamic response speed of the ISNA-PID is quicker, while the overshoot value is also significantly decreased. During transient period, the peak overshoot of the ISNA-PID is 22.34% compare with that of 28.60% for SNA-PID system. The settling time of the ISNA-PID system is attained in very short time, i.e., 0.012 s lower than that of the classical SNA-PID system. Table I shows the comparison of the control system parameters of both controllers when a sinusoidal input signal is fetch into the system. Clearly from this table that the new single neuron adaptive PID controller outperformed the characteristics of the classical single neuron adaptive PID system.

In the third experiment, an additional disturbance is injected to the model system, and change (17) to be:

$$y(k + 1) = 0.368y(k)^2 + 0.264y(k - 1) + 0.632u(k) + d(k)$$

where $d(k)$ is an external disturbance at the 100th sampling instants, with $d(100) = 0.05$. Simulation results on using an external disturbance in addition to the step function in the second experiment are depicted in Figure 7.

As can be seen from this figure, when the external disturbance is fetched to the system, such shown in the bottom of this figure, both the SNA-PID and the ISNA-PID controller systems have the capability to track back the setting value successfully. It is clearly visible in the upscale picture, however, the ISNA-PID controller system shows quicker response speed (rise time), lower settling time, and lower overshoot parameters. As can be seen in Table II, it is numerically proofed that the control system parameters of the

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Rise time (s)</th>
<th>Settling time (s)</th>
<th>Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical SN-APID</td>
<td>0.00134</td>
<td>0.014</td>
<td>28.60</td>
</tr>
<tr>
<td>Improved SN-APID</td>
<td>0.00133</td>
<td>0.012</td>
<td>22.34</td>
</tr>
</tbody>
</table>
Table II. Control system parameters for disturbance.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Rise time (s)</th>
<th>Settling time (s)</th>
<th>Overshoot (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classical SN-APID</td>
<td>0.0044</td>
<td>0.018</td>
<td>5.95</td>
</tr>
<tr>
<td>Improved SN-APID</td>
<td>0.0032</td>
<td>0.013</td>
<td>5.85</td>
</tr>
</tbody>
</table>

ISNA-PID controller system outperformed the classical SNA-PID controller system.

5. CONCLUSIONS

The single neuron adaptive PID controller is a kind of controller that merges the advantage of neural network and the classical PID controller which shows learning ability to track a reference signal with good robustness characteristics and easy on its implementation. In this paper, an improved single neuron adaptive PID controller is developed based on an additional error function derived by the difference between the input control signal and the inverted-control signal through an Inverse Neural Network. Characteristics performance of this novel single neuron adaptive PID controller is then analyzed and compared with that of the classical single neuron adaptive PID controllers. Results of the simulation show that the improved single neuron adaptive PID controller system outperformed the classical one with faster convergence speed, improve dynamic response speed, and decrease the overshoot value. As a conclusion, the proposed adaptive controller system has strong ability for online self-learning with high robustness characteristics.

Acknowledgments: This research was supported by the Universitas Indonesia Research Grant No. 1870/UN2.R12/HKP. 05.00/2015.

References and Notes

2. L. Reznik, O. Ghanayem, and A. Boumisterov, Engineering Applications of Artificial Intelligence 13, 419 (2000).