Design of neural network and PLC-based water flow controller

Burhanuddin Ahmad and Prawito Prajitno*
Department of Physics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Depok, Indonesia

*prawito@sci.ui.ac.id

Abstract. Flow rate is a fundamental physical quantity in the fluid transportation system from one place to another. To achieve this, a reliable controller that is able to produce a constant flowrate in industry is needed. The most used flow controllers in industries are PID-based controllers that are implemented using PLCs. However, there are still shortcomings, they can perform poorly in some applications, for example in the highly nonlinear system which cannot be overcome by conventional PID controllers. There are some other limitations of PID controller, such as PID has the overshoot and undershoots in the output of controlled system and PID gives late response in this study, a neural network-based flow controller is proposed to deal with that problems. The controller will be operated in a miniature plant which consists of a water tank, water pump, a control valve, and a flow transmitter. Due to PLC limitation that cannot be programmed with common programming languages such as MATLAB, a personal computer (PC) is used to run the proposed neural network controller. The PC communicates with the PLC using OPC (OLE for Process Control) server, while the PLC reads the flow transmitter and also controls the control valve directly based on the result output of the neural network controller. In order to evaluate the performance of the proposed controller, several experiments have been conducted. The performance of the proposed controller has been compared with the conventional PID controller. It shows that neural network-based controller outperformed the conventional PID controller, in terms of maximum overshoot and steady-state error, where the neural network controller has maximum overshoot = 5.36% and steady-state error = 0.85%, while the PID controller has 11.3% for overshoot and 1.10 % for steady-state error.

1. Introduction
PID control offers the simplest and yet most efficient solution to many real-world control problems. It is report that more than 90% of industrial controller are still implemented based around PID algorithm [1]. This PID control parameters consists of three types, namely Proportional term (K_p), Integral term (K_i), and Derivative term (K_D). All the three parameters can be adjustable and used together or individually such as Proportional, Proportional Derivative (PD) and Proportional Integral (PI) controllers, depending on the response we want to a plant. The transfer function of series PID controller is shown by equation 1

\[ H(s) = K_p + \frac{K_i}{s} + K_Ds \]  

(1)
However, in the term of implementation to the plant there is lack of understanding on tuning procedures which means the methods of adjusting the parameter PID controller sometimes have difficulty to determine which value is suitable for those three PID control parameters. Determining the PID control parameters is very important things for control the whole process because it affects on performance factors at transient response and the error of the system. Neural Network (NN) has become tremendously popular in the control application due to its ability in adaptive learning based on data and approximating function [2]. It is one of the artificial representations of the human brain that always tries to simulate the learning process as in the human brain [3]. Research on neural network as a controller still continuing to develop and it has been proven that neural network is able to be a controller that can minimize errors and to estimate the output value under different conditions based on the identification of the process at linear as well as nonlinear system by implementing the inverse NN and back-propagation algorithm as a training method [4]. Furthermore, the researchers implemented a NN, as a controller in the tank heating system which is stirred continuously. In this system which is very susceptible to interference and has high sensitivity of process dynamics, the researchers used a combination of the Adaptive Linear Element (Adeline) approach with NN-based controller as an optimization variable to hold disruptions and problems that occur in non-linear systems [5].

Besides being used as a single controller, NN can also be used combined with conventional PID controllers as done in [6]. In that research study, the PID tuning parameter as a NN input quantity of the network that will be optimize and the results will be used to control piston movement and compensate for the effect of congenital pressure in a way implementing the inverse neural network model. In industrial implementation, NN has also been used for predicting errors and monitoring the work of PLC. NN algorithm is used to analyze the received signal processing from the sensors and the execution process of the PLC program. Through a monitoring process of the PLC program, the possibility of errors that will occur can be seen so that it can be detected easily [7].

Based on the successful implementation of NN in controllers that presented above, in this research study a NN-based controller is proposed to control the water flow rate in a prototype plant that uses an accurate low-cost flow transmitter and a control valve. The objective of this study is on NN development to become a controller based on a first-order with dead time plant model. The development begins with the use of Ziegler-Nichols method in open-loop system to construct a PID controller in the prototype plant. Based on the available sequential process variables (PV) data in conjunction with their set-points (SP) data and the associated manipulated variables (MV) data, an NN Model was then built to predict manipulated variable (MV) to control the flow rate of the plant.

2. Design and Implementation

2.1. Lab scale experimental setup

The design of this system consists of hardware and PVC pipes ½ inch in diameter as the process connection. The length of the pipeline used in the system ± 15 meters from the tank to back to the tank. A Process and Instrumentation Diagram is visualized in Figure 1 corresponds to the maintenance and modification of the pipelines transport system for envisioning the physical sequences of equipment used in the lab-scale experimental setup as shown in Figure 2.

The lab scale set up has some transmitter include pressure transmitter (PT001) and flow transmitter (FT001). At the end equal percentage flow characteristic control valve (CV) is implemented and flow transmitter (FT) has been placed. The controller section includes flow indicating controller (FIC) to control the control valve as manipulated variable (MV). PLC mounted FIC decides the level of control valve opening which this FIC is not accessible by the operator which is tuned by PLC-PID controller for operation in the pipeline transport system.
2.2. Programming end communication setup

Programming setup in this project consists of two parts, there are the design of the ladder diagram in the Speed PLC software to program the PLC, and the design of Matlab/SIMULINK to apply the developed NN controller such as in Figure 4. This software design influences each other because the Matlab software requires address as a reference address on the PLC to acquire measured data on the flow transmitter and pressure transmitter. This data will be processed using the developed NN algorithm in the Matlab software.
For implementing the NN algorithm to PLC we need an interface or medium for connecting between PLC and MATLAB as shown in Figure 3. The writer used OPC server to connect it. OPC Server (Object Linking Embedded for Process Control Server) is interface software using a client / server mode based on COM (Component Object Model) / Distributed Component Object Model) that allows MATLAB to communicate with the PLC. The advantages of OPC servers in the process of integrating between PLC and MATLAB directly, which can reduce the level of complexity of the system, facilitate the plant simulation process, reduce additional hardware on the PC, and reduce costs [7].

![Figure 4. NN Control SIMULINK model](image)

### 2.3. Developing Neural Network Controller

The NN learning model that is used in building the NN controller in this research study is shown in Figure 5. The training data are obtained from the MATLAB simulation on the previous PID control that is used as the model desired. The data used for the training process are PV (k-1), PV (k), and SP data and MV (k) data as target data. NN learning will minimize the error between the PV (k+1) of the Model and PV (k+1) generated from the plant. The index k in the variable means that data is taken at the time sampling k, while (k-1) means that data is taken previously.

After training the NN, then the developed neural network models used as a controller in the system as shown in Figure 5. The Matlab function gensim(output) is used to generate the SIMULINK block as shown in Figure 4 [8] [9].

![Figure 5. Block diagram of NN model learning](image)
3. Result and discussion

Table 1. Performance analysis of PID Controller

| Set Point | Max Overshoot | Error | Settling Time | Rise Time |
|-----------|---------------|-------|---------------|-----------|
| 70%       | 7%            | 1.1%  | 9.24s         | 7.26s     |
| 50%       | 15.310%       | 1.09% | 18.36s        | 5.16s     |
| 80%       | 11.836%       | 1.13% | 10.38s        | 8.22s     |

Figure 6. Performance comparison between PID and NN-based controller

Table 2. Performance analysis of NN-Controller

| Set Point | Max Overshoot | Error | Settling time | Rise time |
|-----------|---------------|-------|---------------|-----------|
| 70%       | 4.6%          | 0.8%  | 9.84s         | 8.46s     |
| 50%       | 5.36%         | 0.85% | 8.4s          | 8.22s     |
| 80%       | 4.29%         | 0.08% | 11.4s         | 9.96s     |

By comparing the performance of the two controller above as shown by Table 1 and Table 2, it can be concluded that NN-based controller has better performance than the conventional PID controller, especially in the terms of its maximum overshoot and its steady-state error, where in the average NN-based controller can reduced the maximum overshoot of the plant by 5.94%, as well as in term of steady-state error NN-based controller can make it lower. In terms of the other performance parameters, both controllers have almost the same performance.

It can be seen from the results of the two controller as shown in Figure 6, that the flow transmitter gives fluctuation flow reading. It is due to the flow of the water flow in the pipe of the real plant and also the flow transmitter that has low accuracy and low precision but it shows that the proposed NN-based controller capable of successfully handling this uncertainty by giving better performance in controlling the flow-rate. This result is consistent with the results that have been reported in [2] [8].

4. Conclusion

From this study, after experimenting, collecting and analyzing data, it can be concluded that NN-based controller has better performance than PID controller in terms of maximum overshoot and steady-state error. This result shows that NN is able to deal with the uncertainty (noise) of the plant because of a low accuracy and precision sensor used in it.
Acknowledgment
This work is supported by research grant of Indexed International Publication of Student Final Project (Hibah Publikasi Internasional Terindeks untuk Tugas Akhir (PITTA) Mahasiwa), Universitas Indonesia, Grant No. NKB0657/UN2.R3.1/HKP.05.00/2019.

References
[1]. Ang, Kiam Heong, Gregory Chong, and Yun Li. "PID control system analysis, design, and technology." IEEE transactions on control systems technology 13.4 (2005): 559-576.
[2]. May, Zazilah, Muhammad Hanif Amaran, and Nor Aisyah Ghazali. "Neural network predictive controller for pressure control." Proceedings of the 10th WSEAS international conference on Computational Intelligence, Man-Machine Systems and Cybernetics, and proceedings of the 10th WSEAS international conference on Information Security and Privacy. World Scientific and Engineering Academy and Society (WSEAS), 2011.
[3]. Liu, Jinkun. Intelligent Control Design and MATLAB Simulation. Singapore: Springer Singapore, 2018.
[4]. Priyanka, E. B., C. Maheswari, and S. Thangavel. "Online monitoring and control of flow rate in oil pipelines transportation system by using PLC based Fuzzy-PID Controller." Flow Measurement and Instrumentation 62 (2018): 144-151.
[5]. Priyanka, E. B., C. Maheswari, and B. Meenakshipriya. "Parameter monitoring and control during petrol transportation using PLC based PID controller." Journal of applied research and technology 14.2 (2016): 125-131.
[6]. S. ANUSHA, G. KARPAGAM & E. BHUVANESWARII. “Comparison of Tuning Methods of Pid Controller.” BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE) 2.8 (2014): 1–8. BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE). Web.
[7]. Lieping, Zhang, Zeng Aiqun, and Zhang Yunsheng. "On remote real-time communication between MATLAB and PLC based on OPC technology." 2007 Chinese Control Conference. IEEE, 2007.
[8]. Vamvoudakis, K. G., F.L. Lewis, and Shuzhi Sam Ge. “Neural Networks in Feedback Control Systems.” Mechanical Engineers’ Handbook. John Wiley & Sons, Inc., 2015. 1–52. Mechanical Engineers’ Handbook. Web.
[9]. Demuth, Howard, M Beale, and M Hagan. “Neural Network ToolboxTM 6.” User’s guide (2008): n. pag. User’s guide. Web.
[10]. “Process Control.” Mineral Processing Design and Operations. Elsevier, 2016. 763–816. Mineral Processing Design and Operations. Web.