

Design of Deaerator Storage Tank Level Control System at Industrial Steam Power Plant with Comparison of Neural Network (NN) and Extreme Learning Machine (ELM) Method

W. P. Mahardhika⁽¹⁾, A. Soeprijanto⁽²⁾, M. Syaiin⁽³⁾, S. Wibowo⁽⁴⁾, R. Kurniawan⁽⁵⁾, B. Herijono⁽⁶⁾, R.Y. Adhitya⁽⁷⁾,
E. A. Zuliari⁽⁸⁾, D. K. Setiawan⁽⁹⁾, N. Rinanto⁽¹⁰⁾, B. S. Kaloko⁽¹¹⁾

^(1,3-7,10) Politeknik Perkapalan Negeri Surabaya, ⁽⁸⁾Institut Teknologi Adhi Tama Surabaya, ^(9,11)Universitas Jember,
⁽²⁾Institut Teknologi Sepuluh Nopember

matt.syaiin@gmail.com^(1,3), ryanyudhaadhitya@gmail.com⁽⁴⁻⁷⁾, efrita.zuliari@gmail.com⁽⁸⁾,
dedy.kurnia@unej.ac.id⁽⁹⁾, adisupits@gmail.com⁽²⁾

Abstract - This paper has proposed a prototype of a control system in deaerator storage tank level at Industrial Steam Power Plant based on artificial intelligence (AI). There are two kind methods of AI which are implemented in this research, first is Back Propagation Neural Network (BP-NN) and the second is Extreme Learning Machine (ELM). The proposed method is aimed to improve the performance of an existing Proportional Integral (PI) control method. The input variables are error level and load condition. The output variables are control valve percentage and indicator value. From the experiment, the result proved that ELM is fast superior to BP-NN according to the time of training process and error tolerance. Prototype-based on ELM is also working properly with an error tolerance of 0.15 %.

Keywords : Deaerator, Extreme Learning Machine (ELM), FS-400a, Neural Network (NN), Arduino

I. INTRODUCTION

In Indonesia, the steam power plant industry is growing rapidly. This is due to the electrical energy is one of the important energy that required for human needs. Almost the whole human-equipments are using electrical energy.

One of the most important systems in the process of a steam power plant is a feed water system. And the feed water is stored in the deaerator storage tank, so the deaerator plays an important role in the boiler feed water system[1]. Since the deaerator is a feed water holding tank, the deaerator storage tank level has to be controlled. Control valve is used to adjust the amount of water that goes into the deaerator storage tank so the level can be kept in constant[2]. In the other hands the implementation of the control method in engineering area is growing so fast, not only the methods based on numerical analysis[3-7], but also the methods based on artificial intelligence[8-16].

In this paper, ELM is used as the control method for the valve to keep the deaerator storage tank level in constant condition. The Extreme Learning Machine method is a feed-forward neural network (FNN)[17]. ELM method has an extreme ability in learning speed and has a better accuracy[18]. To verify the performance of ELM control system, BP-NN control system is used as a comparison method. Implementation of ELM as a control method of deaerator storage tank level is a novel method. For an example, several methods that have been implemented in deaerator storage tank control like PID method has some difficulties to get a good control system due to a strong coupling between deaerator pressure and water level[19]. Fuzzy PID Control method, this method is flexible and easily adaptable so the results are work well, static and the dynamic performance is very good, strong resistance and able to adapt itself. And the disadvantages of this method is for getting a good accuracy so the rules of fuzzy must be very precise and the value of KP, KI, KD in PID must also be precise[1]. PID Neural Network method, this method has the advantages of the PID method and also has an advantage in terms of learning and ability, as well as several nonlinear approaches. And the disadvantages of this method are less precision and have a long training time[20]. Besides the advantages of each method mentioned above, the disadvantages of all these methods can be solved by ELM.

The organization of the rest of the paper is as follows. In Section II, the description of research methods and the processing data will be explained. In Section III is about how the ELM method is implemented on hardware. Finally, a conclusion is given in Section IV.

II. METHODOLOGY

The purpose of this research is to create a prototype design of control system of deaerator storage tank control level in an industrial steam power plant with a comparison of Neural Network (NN) and Extreme Learning Machine (ELM) method.

A. System Design

This section describes the system design for this research. The things described are software design, hardware design, software & hardware integration in the prototype.

• Stage 1

In stage 1 is contained how to trial many data in the software, sensor & actuator calibration using Arduino IDE, training & testing validation of the ELM or NN method using MATLAB and monitoring & calculation process of ELM or NN method using Visual Studio 2013. The calculation process of both methods are not directly using arduino mega 2560 but by using Visual Studio 2013, it because arduino has a disadvantage of having a length limit of variable data.

• Stage 2

After the trial procedure of the software has been succeeded, it will continue to Stage 2. In stage 2 will describe how to make and design the hardware. The design of hardware can be seen in figure below.

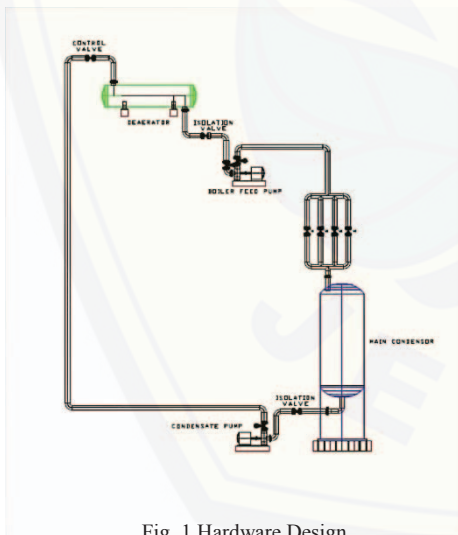


Fig. 1 Hardware Design

In this prototype consist of two tanks, two pumps and some PVC pipe to make the mechanics. For the controller is using Arduino Mega 2560. And for sensors and actuators are the level sensor as a water level detector in the deaerator storage tank, two FS-400a flow sensors as a flow detector into and out of the deaerator storage tank, stepper motor in the clutch

with faucet as a control valve that controls the amount of water into the deaerator storage tank, indicator light and relay module.

• Stage 3

Then when the design of hardware has been succeeded, Stage 3 is started. In stage 3 contained the integration of software and hardware. The working principle of this research are:

1. In the initial condition, all pumps are set off.
2. Then, before the start there are several requirements that must be completed, the condenser tank must be filled with water. If the condenser tank is empty so it must be filled first.
3. After the condenser tank is filled, then the start button is activated.
4. When the start button is active, then both pump turns on.
5. When the pump has been activated, the process of filling the deaerator storage tank occurred.
6. Then the filling process is done until the level of the deaerator storage tank in accordance with the terms is specified.
7. When the level of deaerator storage tank is still not appropriate, there is some condition that should be controlled.
8. The condition is controlled using ELM or NN, it is used to set the percentage of the opening valve. For more information can be seen in table 1.
9. If the system has not been completed, the system will re-fill the deaerator storage tank until the level is specified.
10. If the system has been completed then the stop button is active and makes the both pump is off and the system is finish.

Table 1. Table of ELM or NN Process

INPUT	CONDITION	OUTPUT
ERROR LEVEL	$10 \leq X \leq -10$	OPENED TO 100 % %ORCLOSED TO 0 %
LOAD CONDITION	0% ; 25% ; 50% ; 75% ; 100%	AND INDICATOR HIGH (6), SAFE (5), LOW (4)

B. Application of Neural Network Method on the plant

• Neural Network Design

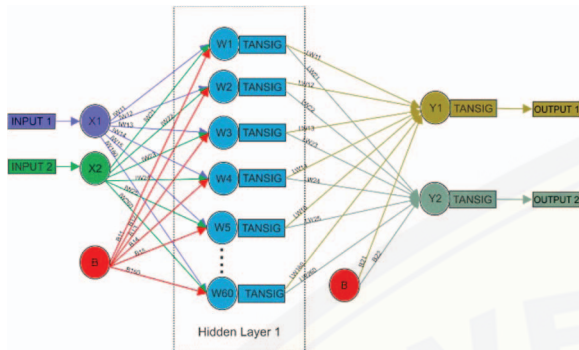


Fig. 2 Neural Network Structure

From Figure 2, it can be explained that the method of Neural Network for this research prototype is using 2 inputs and 2 outputs, 1 hidden layer, 60 neurons in 1st hidden layer, 2 output neurons, and 2 biases. The activation function is using the "TANSIG" function.

• Input and Output Data

Here is the sample of several input and output data for the training process in Neural Network. The data is 80% from 105 actual data.

Table 2. Sample of Training Data

No	Error Level	Load Condition (%)	Control Valve (%)	Indicator
1	-10	0	84	4
2	-9	0	75.5	4
3	-7	0	58.5	4
4	-6	0	50	4
5	-4	0	33	4
6	-3	0	24.5	4
7	-1	0	7.5	4
8	0	0	0	5
...
84	10	100	0	6

• Neural Network Training Process

From the Neural Network training process using Matlab, the performance and regression result for this research are shown in Figure 3 below.

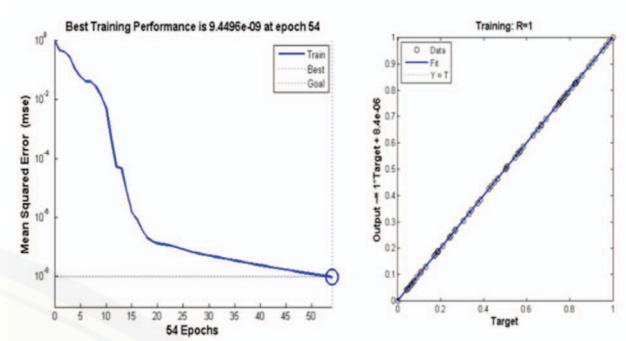


Fig. 3 Best Training Performances and Regression

From the picture above can be explained that the best training performance is 9.4495×10^{-9} and regression was done as much as 54 times from 200 maximum regression. And the training time is 9.53042 seconds.

• Neural Network Training Result

The obtained results from the training process is the weight of input Weight (IW), Layer Weight (LW) and bias. Where the number of IW is 2×60 , the number of LW is 2×60 , the number of bias is 1×60 and 1×2 .

• Neural Network Weight Implementation

After getting the weights, then the next step is to implement the weights into the calculation process in Visual Studio to get the output value. Before it happen, the input is normalized by using equation (1), the process is by dividing the actual data value (a) with the actual data digit length (n). It is used for the value of data training (X) and the normalized result can be reduced between -1 to 1, so the training process is becoming faster.

$$X = \frac{a}{n} \tag{1}$$

Then to find the output value in the calculation process is done by implementing the weights in Visual Studio. The calculation processes are shown in equations (2) to (6).

$$w1 = \text{tansig}((x11 * IW11) + (x2 * IW211) + (1 * B11))$$

$$\vdots \tag{2}$$

$$w60 = \text{tansig}((x160 * IW160) + (x260 * IW260) + (1 * B160))$$

$$oz1 = (w1 * LW11) + \dots + (w60 * LW160) + (1 * B21) \tag{3}$$

$$oz2 = (w1 * LW21) + \dots + (w60 * LW260) + (1 * B22) \tag{4}$$

$$y_1 = \text{tansig}(oz_1) \tag{5}$$

$$y_2 = \text{tansig}(oz_2) \tag{6}$$

After finding the output value 1 (y1) and output 2 (y2) then it will denormalize by doing multiplication between y1 and y2 with the actual data length (n) to get the actual output value.

C. Application of Extreme Learning Machine Method on the plant

• Extreme Learning Machine Design

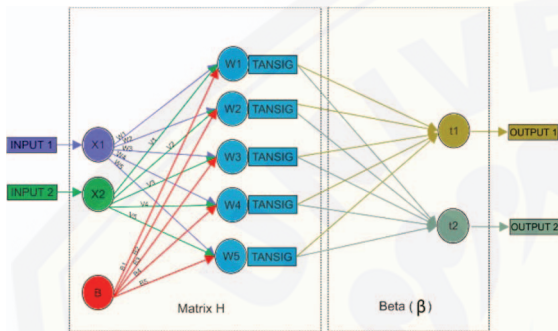


Fig. 4 Extreme Learning Machine Structure

Figure 4 shows Extreme Learning Machine structure for this research prototype is using 2 input and 2 output, H matrix with maximum dimension of 5x5 and minimum of 3x3 and matrix Beta (β). The used activation function is "TANSIG" function.

• Input and Output Data

For input and output of the training data process of ELM method is similar to the table 2 above.

• Extreme Learning Machine Training Process

From the training process of Extreme Learning Machine is done by using Matlab, before it happen the training data must be normalized first. Then in the training process, weights (w, v, b) are determined randomly and calculated in the H matrix. In the training process is also divided into 21 parts of training with the same weight due to the value of similar training data input so the H matrix becomes singular. And the H matrix is used to determine Beta "β".

$$H = \begin{bmatrix} \mathcal{E}(w_1x_{11} + v_1x_{21} + b_1) & \dots & \mathcal{E}(w_sx_1 + v_sx_k + b_t) \\ \vdots & \ddots & \vdots \\ \mathcal{E}(w_1x_n + v_1x_k + b_1) & \dots & \mathcal{E}(w_sx_n + v_sx_k + b_t) \end{bmatrix}$$

For the H matrix (3x3), then the value of n = 13, k = 23, s = 3 and t = 3. And for the H matrix (4x4), then the value n = 14, k = 24, s = 4 and t = 4. Then for the H matrix (5x5) the value of n = 15, k = 25, s = 5 and t = 5. And (E) is tansig activation function.

• Extreme Learning Machine Training Result

The result of training is matrix beta (β) with the size of 5x2. Where :

$$\text{Beta}(\beta) = H^{\dagger}T.$$

And T is the value of training data output.

• Beta (β) Implementation

This section is used to find the value of output training using Visual Studio. To determine the value of training output is using the equation (7). Here is the first H row matrix equation.

$$H = [\mathcal{E}(w_1x_{11} + v_1x_{21} + b_1) \quad \dots \quad \mathcal{E}(w_sx_{11} + v_sx_{21} + b_t)]$$

For the H matrix (1x3), then the value of s = 3 and t = 3. For the H matrix (1x4), the value of s = 4 and t = 4. Then for the H matrix (1x5), the value of s = 5 and t = 5.

$$y = H * \beta \tag{7}$$

III. DATA ANALYSIS AND DISCUSSION

A. Test Scenario or Procedure

In this section explains how the procedure of testing process to prove which one of both methods can work well. The testing process is to compare 5 actual training data with the training data results of NN and ELM. Then the comparison of 5 validation data (in addition to training data) with the actual validation data of NN and ELM. For more details can be seen in figure 5.

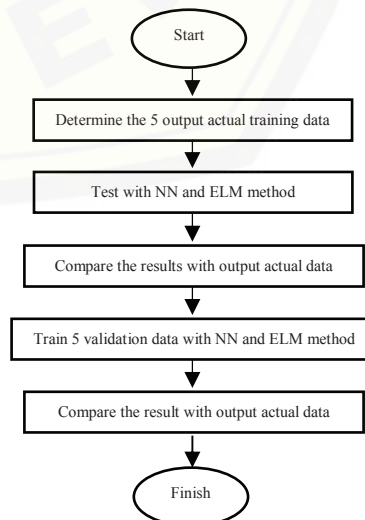


Fig. 5 Flow chart Of Test Scenario and Procedure

B. Comparison of Neural Network Performance and Extreme Learning Machine.

Table 3 is the performance comparison result of Neural Network and Extreme Learning Machine method.

Table 3. NN VS ELM Comparison Result

Training Data						
Input		Output			Error ²	
Error Level	Load Condition (%)	CV (%)	Ind	NN		NN
-9	0	75.5	4	75.50685	4.00712	4.69E-05 5.07E-05
-4	25	45	4	44.99727	4.007084	7.47E-06 5.018E-05
0	50	42.5	5	42.50281	5.011765	7.88E-06 0.0001384
2	75	50	6	49.99305	6.010399	4.83E-05 0.0001081
8	100	37.2	6	37.21039	6.001619	0.000108 2.621E-06
Total Error					0.000219	0.0003501
Root Mean Square Error (RMSE)					0.006612	0.0083674
Validation Data						
Input		Output			Error ²	
Error Level	Load Condition (%)	CV (%)	Ind	NN		NN
-5	0	41.5	4	39.5565	3.817594	3.777204 0.0332718
-3	25	39	4	50.73921	3.769358	137.8089 0.0531957
5	50	22.5	6	18.53428	6.059001	15.72693 0.0034812
8	75	11	6	13.03489	5.940531	4.140783 0.0035366
6	100	74.4	6	74.91622	5.806361	0.266483 0.0374959
Total Error					161.7203	0.1309812
Root Mean Square Error (RMSE)					5.687185	0.1618525
Training Data						
Input		Output			Error ²	
Error Level	Load Condition (%)	CV (%)	Ind	ELM		ELM
-9	0	75.5	4	75.5	4	1.55E-17 1.051E-22
-4	25	45	4	45	4	3.85E-19 1.456E-17
0	50	42.5	5	42.50004	4.999997	1.95E-09 1.121E-11
2	75	50	6	50	6	1.41E-15 3.323E-17
8	100	37.2	6	37.2	6	1.34E-17 7.898E-20
Total Error					1.95E-09	1.121E-11
Root Mean Square Error (RMSE)					1.98E-05	1.497E-06
Validation Data						
Input		Output			Error ²	
Error Level	Load Condition (%)	CV (%)	Ind	ELM		ELM
-5	0	41.5	4	41.50001	4	6.55E-11 2.025E-15
-3	25	39	4	39	4.254436	1.02E-11 0.0647376
5	50	22.5	6	22.50001	6	2.04E-10 1.115E-16
8	75	11	6	10.99969	6	9.42E-08 1.483E-15
6	100	74.4	6	74.40001	6	1.71E-10 2.09E-15
Total Error					9.47E-08	0.0647376
Root Mean Square Error (RMSE)					0.000138	0.1137871

From table 3 above shows Root Mean Squared Error (RMSE) value on the NN method in training data is smaller than validation data. And for RMSE in ELM method in training data is also smaller than validation data. However, if the RMSE is overall compared between the NN and ELM methods, the RMSE value of the ELM method is smaller than the NN method. Training time for the NN method is longer than the average training time of 21 parts of the ELM training method. For more details, the

performance comparisons of NN and ELM are shown in table 4 below.

Table 4. NN VS ELM Performance Comparison Result

No	Description	NN		ELM	
		Output 1	Output 2	Output 1	Output 2
1	Training RMSE	0.006612	0.008367	1.98E-05	1.50E-06
2	Validation RMSE	5.687185	0.161853	0.000138	0.113787
3	Training Time	9.539042 second		0.01317319 second	

IV. CONCLUSION

Based on the experimental experience in this research can be concluded that ELM is better and superior when compared with NN. Based on the time of training, ELM is very fast compared with NN. ELM accuracy in training and validation results has an average error smaller than NN predicted outputs. So to recognize patterns and variations of the new data, ELM is able to provide predictions that are almost identical to the target data. Because in deaerator storage tank level control requires great accuracy and high adaptation of time efficiency, the ELM method can be made as the primary choice rather than the NN method and other methods. Thus, the danger that arises from the failure of the deaerator storage tank level control can be avoided.

REFERENCES

- [1] Jinli, L. *Adaptive Fuzzy PID Control for Boiler Deaerator*. in 2012 International Conference on Industrial Control and Electronics Engineering. 2012.
- [2] Geng, L. *A class of model reference adaptive decouple control based on RBF neural network in deaerator system*. in 2008 3rd IEEE Conference on Industrial Electronics and Applications. 2008.
- [3] Syai'in, M., K.L. Lian, and T.-H. Chen. *Improved Robustness of Sequential Three Phase Power Flow Using Homotopic Method*. 2013, 2013. 8(1).
- [4] Syai'in, M. and L. Kuo Lung. *Microgrid power flow using Homotopic and Runge-Kutta Method*. in 2015 IEEE 2nd International Future Energy Electronics Conference (IFEEEC). 2015.
- [5] Lian, K.L. and M. Syai'in. *Steady-state solutions of a voltage source converter with dq-frame controllers by means of the time-domain method*. IEEE Transactions on Electrical and Electronic Engineering, 2014. 9(2): p. 165-175.
- [6] Syai'in, M., O. Penangsang, and A. Soeprijanto. *Real-time unbalanced load flow development using direct-Z<inf>BR</inf> method and modified Lambda iteration for on-line monitoring and control*. in 22nd International Conference and Exhibition on Electricity Distribution (CIRED 2013). 2013.
- [7] Lian, K.L., et al. *Robust microgrid power flow using particle swarm optimization*. in International School on Nonsinusoidal Currents and Compensation 2013 (ISNCC 2013). 2013.
- [8] Aryani, N.K., et al., *Optimal Placement and Sizing of Distributed Generation for Minimize Losses in Unbalance Radial Distribution Systems Using Quantum Genetic Algorithm*. 2014, 2014. 9(1).
- [9] Syai'in, M., K.L. Lian, and A. Soeprijanto. *Digital Generator Capability Curve for Improving Optimal Power Flow based on IPSO*. 2013, 2013. 8(2).
- [10] Syai'in, M., et al. *A distribution power flow using particle swarm optimization*. in 2012 IEEE Power and Energy Society General Meeting. 2012.
- [11] Syai'in, M. and A. Soeprijanto. *Neural Network Optimal Power Flow (NN-OPF) based on IPSO with Developed Load Cluster Method*. in World Academy of Science, Engineering and Technology. 2010.
- [12] Adhitya, R.Y., et al. *Comparison methods of Fuzzy Logic Control and Feed Forward Neural Network in automatic operating temperature and humidity control system (Oyster Mushroom Farm House) using microcontroller*. in 2016 International Symposium on Electronics and Smart Devices (ISESD). 2016.

- [13] M.Syai'in and A.Soeprijanto, *Combination of Generator Capability Curve Constraint and Statistic-Fuzzy Load Clustering Algorithm to improve NN-OPF performance*. Journal of Electrical Systems, 2012. **08**(02): p. 198-208.
- [14] Rinanto, N., et al. *Rotor bars fault detection by DFT spectral analysis and Extreme Learning Machine*. in *2016 International Symposium on Electronics and Smart Devices (ISESD)*. 2016.
- [15] Syai'in, M., et al. *Smart-Meter based on current transient signal signature and constructive backpropagation method*. in *2014 The 1st International Conference on Information Technology, Computer, and Electrical Engineering*. 2014.
- [16] M.Syai'in, et al., *Incremental Particle Swarm Optimizer with local search for Optimal Power Flow Subjected to Digital GCC based on Neural Network*. International Journal of Digital Content Technology and its Applications (JDCTA), 2012. **07**(06): p. 242-252.
- [17] Lin, S., et al., *Is Extreme Learning Machine Feasible? A Theoretical Assessment (Part II)*. IEEE Transactions on Neural Networks and Learning Systems, 2015. **26**(1): p. 21-34.
- [18] Agustina, I.D., W. Anggraeni, and A. Mukhlason, *Penerapan Metode Extreme Learning Machine untuk Peramalan*. ITS Undergraduate 9832 Paper, 2010: p. 1.
- [19] Peng, W., et al. *Decoupling control based on PID neural network for deaerator and condenser water level control system*. in *2015 34th Chinese Control Conference (CCC)*. 2015.
- [20] Wang, P., H. Meng, and Q. Ji. *PID neural network decoupling control of deaerator pressure and water level control system*. in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*. 2014.

