A Comparison Study of MIMO Water Wall Model with Linear, MFNN and ESN Models

Un-Chul Moon[†], Jaewoo Lim* and Kwang Y. Lee**

Abstract – A water wall system is one of the most important components of a boiler in a thermal power plant, and it is a nonlinear Multi-Input and Multi-Output (MIMO) system, with 6 inputs and 3 outputs. Three models are developed and compared for the controller design, including a linear model, a multilayer feed-forward neural network (MFNN) model and an Echo State Network (ESN) model. First, the linear model is developed by linearizing a given nonlinear model and is analyzed as a function of the operating point. Second, the MFNN and the ESN are developed by using training data from the nonlinear model. The three models are validated using Matlab with nonlinear input-output data that was not used during training.

Keywords: Water wall model, Power plant modelling, Power plant identification, Linearization, Multilayer feed-forward neural network, Echo state network

1. Introduction

Thermal power plants supply approximately 65% of the world's electric power [1], and a water wall system consists of a collection of metal tubes in the furnace of a thermal power plant. Recirculating water in the water wall is heated by combustion energy, and its phase changes from water to steam. Therefore, it is very important to properly model, control and analyze the water wall system in order to manage the entire boiler-turbine system [2].

Several principles, including the energy balance, mass balance and thermal equilibrium, can be used to explain the dynamics of the water wall system [3]. As a result, the water wall system is modeled as a dynamic nonlinear multi-input multi-output (MIMO) system. However, the controller for a nonlinear system is usually designed using a linear model based on a particular operating point. Therefore, the degree of nonlinearity of a water wall system needs to be investigated to adequately apply a linear control system.

If the object system has a severe nonlinearity or a wide operating range, the linear model will have a limitation in its ability to describe the dynamics of the object system [4]. An alternative for this kind of nonlinear control problem involves implementing a neural network for the modeling and controller design [5]. Since it is difficult to develop a reliable mathematical model of the nonlinear dynamics in practice, the neural network is typically trained according to the measured input-output data.

Therefore, a mathematical nonlinear dynamic model is not necessary for this approach.

The multilayer feedforward neural network (MFNN) has been successfully applied to model static nonlinear systems [6]. However, it requires a number of tapped delays of the output neurons as input signals for dynamic systems, resulting in an increased number of input neurons and an extensive training time. Recurrent neural networks (RNN) with feedback connections are potential approaches that better represent dynamic systems, including Elman networks [7] and diagonal recurrent neural networks (DRNN) [8]. However, training for recurrent neural networks is relatively more complex than that for MFNN [9].

Recently, eco state network (ESN), a special kind of three layer RNN, was proposed by Jaeger [10]. The basic idea is to use a large "dynamic reservoir" to supply interesting dynamics from which the desired output is combined. One of the most interesting things of ESN is that it can be trained in a one-shot fashion without repetitively passing through the training set that is usually required to train other RNN [11].

In [12], Jaeger and others used the ESN to optimize problems and demonstrated simple applications. Lin and others used ESN to predict the stock price and to suggest stock trades [13]. Ishii and others used ESN to identify the yawing acceleration of an underwater robot [14]. Dai and others used ESN to predict the harmonic current from nonlinear loads [15]. Pan and Wang proposed a model for predictive control based on recurrent neural network, where they used ESN to identify an unknown nonlinear dynamic system [16].

In this paper, three models, including a linearized model, a MFNN model and a nonlinear ESN model, are developed and compared from the point of view of controller design for a practical water wall system in a 600 MW oil drum

† Corresponding Author: School of Electrical and Electronic Engineering, Chung-Ang University, Korea. (ucmoon@cau.ac.kr)
* School of Electrical and Electronic Engineering, Chung-Ang University, Korea. (jaewoo1129@naver.com)
** Dept. of Electrical and Computer Engineering, Baylor University, Waco, TX 76798, USA. (Kwang_Y_Lee@baylor.edu)

boiler-turbine system. The three models are developed and compared using Matlab on a personal computer environment. First, we introduce the nonlinear water wall model with the energy balance, mass balance and thermal equilibrium equation. Then, the linear model is developed with a Taylor series expansion. The changes in the pole and steady state gain according to the change in the operating point are introduced. To train the MFNN and the ESN, Pseudo Random Binary Noise Signal (PRBNS) inputs are applied to the nonlinear water wall model. After training the MFNN and ESN, the performance of the linear model, MFNN and ESN is evaluated, presented and discussed.

2. A Water Wall Modeling

2.1 A nonlinear water wall model

In this paper, we consider a water wall model in a 600MW oil drum boiler-turbine system [3]. The water wall is located between the recirculating pump and the drum in the boiler. The circulating water in the water wall is heated by combustion in the furnace, and its phase changes from water to steam, and then it is poured into the drum.

We assume that the water wall system is a nonlinear model, as used by Usoro [3]. The mass balance equation, energy equilibrium equation, and thermal equilibrium equation can be defined according to the boiler structure and physical principles. The description of the water wall system is as follows:

Mass balance equation is given by:

$$W_{wwo} = W_{rw} + K_{wrps} \qquad (1)$$

where, W_{wwo} : Mass flow of water wall outlet,
$\quad W_{rw}$: Mass flow of recirculating water,
$\quad K_{wrps}$: Mass flow of recirculating pump.

According to [3], K_{wrps} accounts for recirculating pump leakages and seal injection. It is small and may be neglected.

Energy equilibrium equation is given by:

$$M_{wwme} \times K_{swwm} \times \frac{d}{dt}(T_{wwm}) = Q_{wwgm} - Q_{wwmw} \qquad (2)$$

where, M_{wwme} : Effective mass of water wall metal,
$\quad K_{swwm}$: Specific heat of water wall metal,
$\quad T_{wwm}$: Temperature of water wall metal,
$\quad Q_{wwgm}$: Heat transfer rate of gas to metal,
$\quad Q_{wwmw}$: Heat transfer rate of metal to water.

$$M_{wwme} = K_{mwwm} + \frac{K_{vww} \times R_{drw} \times H_{drw}}{K_{swwm} \times T_{wwm}} \qquad (3)$$

where, K_{mwwm} : Mass of water wall metal,

K_{vww} : Volume of water wall,
R_{drw} : Density of drum water,
H_{drw} : Enthalpy of drum water,

$$Q_{wwmw} = K_{uwwmw} \times (T_{wwm} - T_{drs})^3 \qquad (4)$$

where, K_{uwwmw} : Constant with dimension of $[W/K^3]$,
$\quad T_{drs}$: Temperature of drum steam,

Thermal equilibrium equation of circulating water is given by:

$$W_{wwo} \times (H_{wwo} - H_{rpo}) = Q_{wwmw} \qquad (5)$$

where, H_{rpo} : Enthalpy of recirculating pump outlet,
$\quad H_{wwo}$: Enthalpy of water wall outlet,

The following constant values are used:
K_{mwwm}=1063000, K_{vww}=2318.61, K_{swwm}=0.11, K_{uwwmw}= 173.5205

For (1)-(5), we define the inputs, the outputs, and the states as follows,

$$U = [u_1, u_2, u_3, u_4, u_5, u_6]^T$$
$$= [W_{rw}, H_{rpo}, R_{drw}, H_{drw}, T_{drs}, Q_{wwgm}]^T \qquad (6)$$

$$Y = [y_1, y_2, y_3] = [W_{wwo}, H_{wwo}, T_{wwm}]^T \qquad (7)$$

$$X = [x] = [T_{wwm}] \qquad (8)$$

Then, the nonlinear model of water wall system is represented in the following equation,

$$\dot{x} = \frac{u_6 - K_{uwwmw} \times (x - u_5)^3}{(K_{mwwm} + \frac{K_{vww} \times u_3 \times u_4}{K_{swwm} \times x}) \times K_{swwm}} \qquad (9)$$

$$y_1 = u_1 \qquad (10)$$

$$y_2 = u_2 + \frac{K_{uwwmw} \times (x - u_5)^3}{u_1} \qquad (11)$$

$$y_3 = x \qquad (12)$$

From the above equations, we notice that y_2 and y_3 of the water wall system have severe nonlinearity, while y_1 is u_1, which is a simple linear output.

2.2 Effect of operating point to linear model

The nonlinear model, given by (9)-(12), is linearized using the first-order Taylor approximation at an operating point. We generalized the operating points as follows:
Operating point=$[u_{10}, u_{20}, u_{30}, u_{40}, u_{50}, u_{60}, x_0, y_{10}, y_{20}, y_{30}]$
As an example, the operating point at a power output of 400 MW is shown in Table 1, where steady state values of 6 inputs, 3 outputs, and a state can be observed. When all

Table 1. The operating of water wall

Variables	Operating value
u_1 (W_{rw})	4716.7199
u_2 (H_{rpo})	721.4824
u_3 (R_{drw})	34.4693
u_4 (H_{drw})	733.1763
u_5 (T_{drs})	1132.6393
u_6 (Q_{wwgm})	267289.8666
y_1 (W_{wwo})	4716.7199
y_2 (H_{wwo})	778.0994
y_3 ($=x$, T_{wwm})	1144.1883

the variables are defined as deviations from the operation values in Table 1, the linear model at 400 MW is given as,

$$\Delta \dot{x} = -8.2535 \times 10^{-12} \Delta u_3 - 3.8803 \times 10^{-13} \Delta u_4$$
$$+ 0.4129 \Delta u_5 + 5.9474 \times 10^{-6} \Delta u_6 - 0.4129 \Delta x \tag{13}$$

$$\Delta y_1 = \Delta u_1 \tag{14}$$

$$\Delta y_2 = -0.012 \Delta u_1 + \Delta u_2 - 14.7070 \Delta u_5 + 14.7070 \Delta x \tag{15}$$

$$\Delta y_3 = \Delta x \tag{16}$$

We notice that (13) is a simple first order system with a pole at s = -0.4129, and the overall system is stable without an oscillation mode. The transfer function matrix, $\overline{G}(s)$, of (13)-(16) for the 6-input 3-output system is given by

$$\overline{G}(s) = \begin{pmatrix} 1 & 0 & 0 \\ -0.012 & 1 & \dfrac{-1.214 \times 10^{-10}}{s + 0.4129} \\ 0 & 0 & \dfrac{-8.254 \times 10^{-12}}{s + 0.4129} \end{pmatrix}$$

$$\begin{array}{ccc} 0 & 0 & 0 \\ \dfrac{-5.707 \times 10^{-12}}{s + 0.4129} & \dfrac{-14.71s}{s + 0.4129} & \dfrac{8.747 \times 10^{-5}}{s + 0.4129} \\ \dfrac{-3.88 \times 10^{-13}}{s + 0.4129} & \dfrac{0.4129}{s + 0.4129} & \dfrac{5.947 \times 10^{-6}}{s + 0.4129} \end{array} \tag{17}$$

where G_{ij} is a transfer function of the *i*-th output from *j*-th input. According to (13) and (17), the linear model at 400 MW is a first order system with a pole on the left half s-plane.

To analyze the nonlinear model, the above linearization is performed as a function of the electrical power output. Then, the linear models that are generated at each operating point are compared from the point of view of the pole and the steady-state gain. Fig. 1 shows the movement of the pole with the operating points. The pole at 100 MW is at -0.2335 and moves to the left as the power increases. The pole at 600MW is at -0.6084. This means that the transient is becoming shorter by several times as the load demand increases.

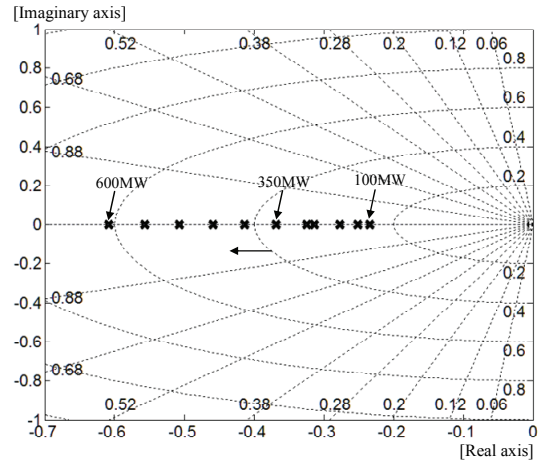Fig. 2 shows the steady-state gain changes of y_2 and y_3 as

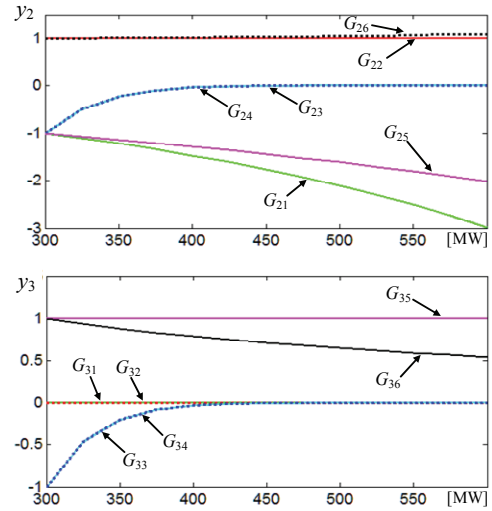**Fig. 1.** Movement of the pole according to operating point

Fig. 2. The change in the steady-state gain for y_2 and y_3 with the operating point

a function of the operating points from 300 [MW] to 600 [MW], which are normalized with a gain at 300 [MW]. The gain for y_1 is not represented because it has no dynamics. In Fig. 2, the steady-state gain changes significantly. For example, the amplitude of the steady-state gain for G_{21} at 600 [MW] negatively increases to three times that of 300 [MW]. From Figs. 1 and 2, even with a good nonlinear water wall model, suitable performance is hard to achieve for wide range operation using a conventional linear control technique.

3. Two Neural Network Models

3.1 PRBNS training data

As an alternative to conventional linear modeling and control, neural networks have been extensively applied to nonlinear modeling and controller design [5]. Due to its
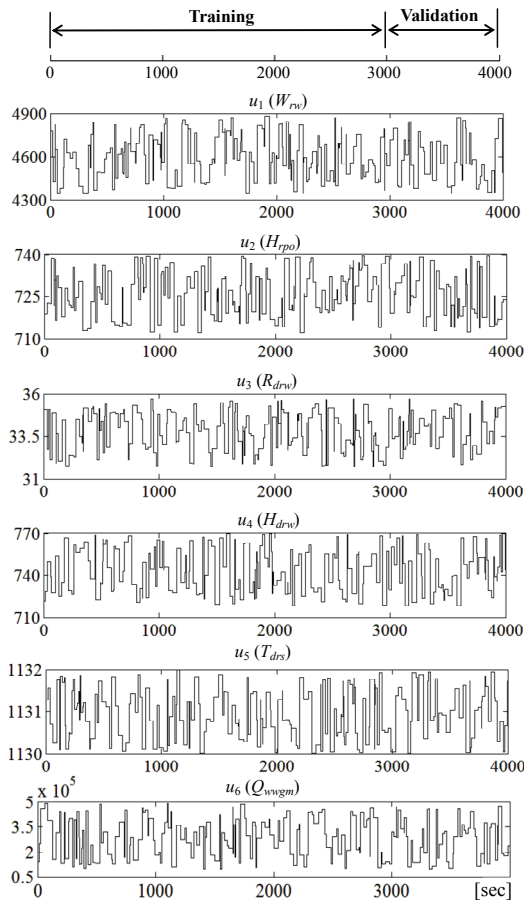
Fig. 3. PRBNS input for nonlinear water wall system

Fig. 4. Output of PRBNS input for nonlinear water wall system

ability to learn nonlinear functions, neural networks do not need a mathematical model of the object system. Instead, the model is developed with the given input-output data. Therefore, it is very important to make a suitable selection of the training data.

To identify the dynamic system, a Pseudo Random Binary Noise Signal (PRBNS) is a popular input signal that is used to excite the system over a wide frequency range [17]. In order to obtain training data for the nonlinear water wall system considered in this paper, the PRBNS for six inputs is generated for 4000 [sec] with 0.1 [sec] sampling time.

Fig. 3 shows generated PRBNS input, and Fig. 4 shows the corresponding output using a nonlinear water wall system in (9)-(12). In these figures, the first 3000 [sec] data are used to train the neural network, and the latter 1000 [sec] data are left for validation. That is, the data between 3000 [sec] and 4000 [sec] is not used for training and is instead used to evaluate the trained neural network model.

3.2 Multilayer feed-forward neural network model

The multilayer feed-forward neural network (MFNN) is a standard neural network [6]. It has been extensively applied with the "error back-propagation" training algorithm
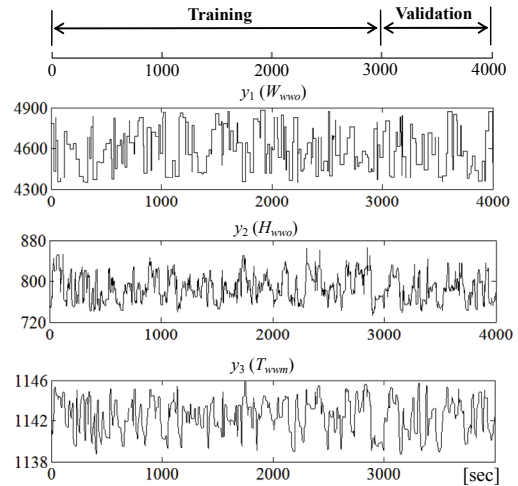
to various nonlinear modeling and control problems [5].

In this paper, MFNN is first used to model a nonlinear water wall system. To train the MFNN, the training data is rearranged to reflect the water wall system of (9)-(12) as follows,

$$Y(n+1) = f\{U(n), Y(n)\} \tag{18}$$

where, n is the discrete time step. And, $U(n)=[u_1(n), \cdots, u_6(n)]$, $Y(n)=[y_1(n), y_2(n), y_3(n)]$ and $Y(n+1)=[y_1(n+1), y_2(n+1), y_3(n+1)]$. The MFNN used in this paper has a three-layer feed-forward neural network, including an input layer, a hidden layer and an output layer. The input of MFNN is $U(n)$ and $Y(n)$, and the number of the input node is 9. The output of the MFNN is $Y(n+1)$, the number of output node is 3.

The purpose of the training is to minimize the total square error between the training output data and the MFNN output data. This is actually an unconstraint optimal problem where the error function is defined as follows,

$$\min_{W} \quad E_{MFNN} = \sum_{n}^{N}(Y_{MFNN}(n) - Y(n))^2 \tag{19}$$

where W represents all the weights. Y_{MFNN} represents the output from the MFNN, Y represents the output for the training data, N represents the number for the training data, which is 30,000 and n is the discrete time step. To consider the different range of each output, each output is normalized in Eq. (19). The optimal problem, (19) is solved using the back propagation algorithm [18]. The Matlab toolbox is used to train the MFNN in this paper.

One of the most important parameters for the MFNN is the number of hidden neurons. Usually, a larger number of hidden neurons decreases the error but increases the training time of the MFNN. Therefore, some kind of
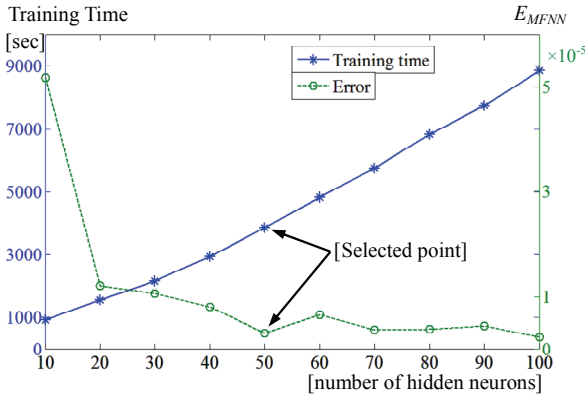
Fig. 5. Training time of MFNN and E_{MFNN} as a function of hidden neuron number

compromise is necessary. In this paper, different numbers of hidden neurons are independently tested. Fig. 5 shows the E_{MFNN} and the corresponding training time for 10 different numbers of hidden neurons, from 10 to 100. From Fig. 5, considering the increase in training time, hidden neurons by more than 50 are redundant to decrease the error. Therefore, in this paper, the number of hidden neurons of the MFNN is selected to be 50 for the nonlinear water wall model, which has 3.00×10^{-6} of E_{MFNN} and 3853.6 [sec] of training time.

3.3 Echo state network model

The ESN is a special form of RNN with three layers: an input layer, a hidden layer and an output layer. The significant characteristic of the ESN is a hidden layer with a large number of neurons that are sparsely and randomly inter-connected and/or self-connected. This is meant to imitate a biological human brain system. The node is commonly referred to as a "dynamic reservoir" that can be excited by connecting it with the input unit and/or the feedback output unit. A more detailed explanation of the ESN was described by Jaeger [10]. Such ESNs have been recently applied successfully for dynamic system identification and control [11, 19-21].

In this paper, the ESN is used as a second neural network model to describe the nonlinear dynamics of the water wall system. Fig. 6 shows the structure of the ESN used in this paper. In the figure, the ESN has 6 inputs and 3 outputs that are the same as those of a water wall system. The connection weights are divided into 4 categories, W^{in}, W^{dr}, W^{fb} and W^{out}, which are the input weight matrix, internal weight matrix, output feedback weight matrix and output weight matrix, respectively.

In this figure, W^{in} is represented with solid lines from the input nodes to the reservoirs; W^{dr} is represented with solid lines among reservoirs; W^{fb} is represented with solid lines from the output nodes to the reservoirs; and W^{out} is represented with dotted lines from the input, reservoirs and output nodes to the output nodes. When the number
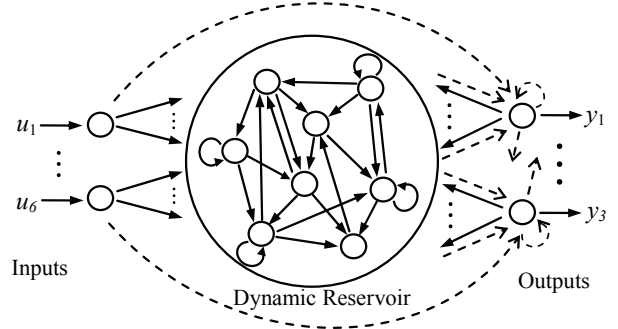

Fig. 6. The architecture of an ESN for a water wall system

of reservoirs is N_{dr}, the dimensions of W^{in}, W^{dr}, W^{fb} and W^{out} are $(N_{dr} \times 6)$, $(N_{dr} \times N_{dr})$, $(N_{dr} \times 3)$ and $(3 \times (N_{dr} + 9))$, respectively.

The state of the reservoir and output from input in Fig. 6 are computed as follows:

$$S(n+1) = f\left(W^{in}U(n) + W^{dr}S(n) + W^{fb}Y(n)\right) \quad (20)$$

$$Y(n+1) = W^{out}\begin{pmatrix} U(n) \\ S(n+1) \\ Y(n) \end{pmatrix} \quad (21)$$

where $S(n)=[s_1(n), \cdots, s_{Ndr}(n)]^T$ is the state of the dynamic reservoir, $U(n)=[u_1(n), \cdots, u_6(n)]^T$ is the input, $Y(n)=[y_1(n), y_2(n), y_3(n)]^T$ is the output, $f(\cdot)$ is the hyperbolic tangent function and n is a discrete time step.

In the ESN, W^{in}, W^{dr} and W^{fb} are fixed with random numbers that were initially generated while W^{out} is trained with the given input-output data. This is a significant difference between ESN and other RNNs, which enable ESN training through a simple linear regression [10, 22]. Randomly generated (W^{in}, W^{dr}, W^{fb}) determine the "echo state properties" which means that "if the network has been run for a long time, the current network state is uniquely determined by the history of the input and the teacher forced output." [12]. A more detailed analysis of the sufficient conditions and necessary conditions for the echo state property are presented by Zhang et al. [23].

To guarantee the echo state properties, W^{dr} is usually generated as follows [16, 19]:

$$W^{dr} = \alpha \frac{W_0^{dr}}{|\lambda_{max}|} \quad (22)$$

where, W_0^{dr} is the initial internal weight matrix with a sparse connectivity of 5% over the range [-0.5, 0.5], and its mean value is about zero, λ_{max} is the highest eigenvalue of W_0^{dr}, and α is a constant of less than one, which is referred to as a spectral radius. Also, W^{in} and W^{fb} are randomly chosen in the range [-1, 1].

The training of the ESN is an off-line calculation of the output weight W^{out} to minimize the error square as follows:

$$\min_{W^{out}} \quad E_{ESN} = \sum_{n}^{N}(Y_{ESN}(n) - Y(n))^2 \qquad (23)$$

where Y_{ESN} represents the output of the ESN and Y represents the normalized output of the training data. In (23), initial part of the training data is not considered into the E_{ESN} since the initial dynamic reservoir, $S(0)$, is initially set with an arbitrary value that affects the output. This part is referred to as the "initial washout time" and is usually selected empirically [10, 15, 16, 19, 22]. In this paper, this time is selected as 500 [sec], and therefore, the number of training data N is 25,000 [step] for ESN.

To minimize the E_{ESN}, the training input $U(n)$, the internal states $S(n+1)$ and the training output $Y(n+1)$ are collected together into a new row of a matrix M. That is, M is the concatenated matrix of U, S and Y, and its size is $N\times(N_{dr}+9)$. At the same time, the training outputs $Y(n)$ are collected into another new matrix T with size ($N\times3$). Then, the desired W^{out}, which minimize the E_{ESN}, are obtained by multiplying the pseudo-inverse of M with T as follows [15, 19, 22].

$$W^{out} = \{M^T(MM^T)^{-1}T\}^T = (M^+T)^T \qquad (24)$$

where, M^+ denotes the pseudo-inverse of the M matrix. Since training in the ESN is simply performed with (24), the training time for the ESN is mainly determined as the time to calculate the pseudo-inverse. In this paper, the Matlab function *pinv*( ) is used to calculate the pseudo-inverse using singular value decomposition. This simple training is a major benefit of the ESN over other neural networks.

In practice, the number of reservoirs (N_{dr}) and the value of the spectral radius (α) are important parameters for the performance of the ESN. Usually, a large N_{dr} supplies various dynamics to decrease the error while increasing the redundancy for the ESN. A small α decreases the inter-connected weights and the self-connected weights of the dynamic reservoirs. Therefore, a small α might be sufficient to describe the dynamics with a small time constant while a large α is required to describe the
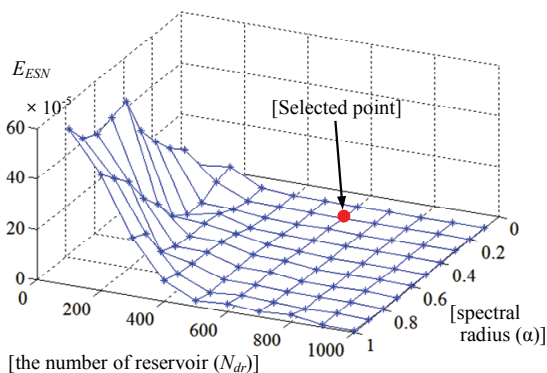

Fig. 7. E_{ESN} as a function of the reservoir number and spectral radius
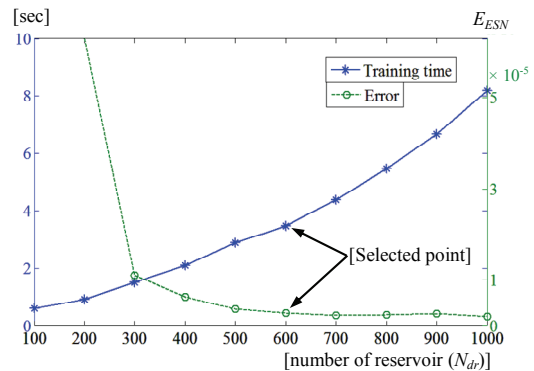

Fig. 8. Training time of the ESN and E_{ESN} as a function of the reservoir number

necessary dynamics with a large time constant [24].

In this paper, different values of N_{dr} and α are tested independently. Fig. 7 shows the E_{ESN} as a function of N_{dr}, and α when N_{dr} is 100, 200, …, 1000 and α is 0.1, 0.2, …, 1. Since the E_{ESN} is dependent on a randomly generated initial weight set (W^{in}, W^{dr}, W^{fb}), the E_{ESN} of each point in Fig. 7 is selected with the best value of ten initial weight sets. From Fig. 7, the ESN shows a small error with a large N_{dr} and small α. From a comparison with the MFNN, an ESN with more than 700 reservoirs and less than 0.2 of α shows a smaller error than that of MFNN. In this paper, the final parameters of the ESN model used to describe the water wall system are selected with 600 for N_{dr} and 0.2 for α, which has a 2.73×10^{-6} E_{ESN}.

Fig. 8 shows the training time of the ESN with respect to N_{dr}. Since α does not affect the calculation time with (24), the training time is represented with N_{dr} at 0.2 of α in this figure. Naturally, a large N_{dr} requires a large training time. In Figs. 5 and 8, the training time for the ESN is noted to be of less than 10 [sec] while that of MFNN is of several thousand [sec]. This is one of the important benefits of ESN, which can be trained in a one-shot manner with (24) to minimize (23). The training time for the selected ESN model is of 3.5 [sec] in this paper.

4. Comparison Results

The performance of the three models, i.e., the linear, MFNN and ESN models, is tested using PRBNS data that were not used during training. To evaluate the performance, the PRBNS input data in Fig. 3 are independently applied to the three models. Then, nonlinear outputs between 3000 [sec] and 4000 [sec] in Fig. 4, which is not used during training, are compared with the three outputs of the MFNN, ESN and linear models.

Fig. 9 shows the comparison of y_1 for the linear, MFNN and ESN models. Since y_1 is a linear output, the linear model is identical to the nonlinear model. In this case, both MFNN and ESN show almost the same response as that of the nonlinear model.
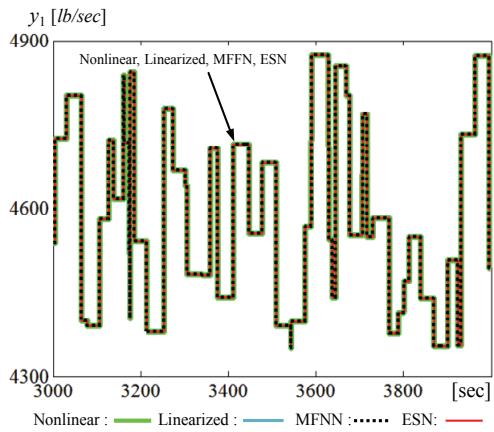
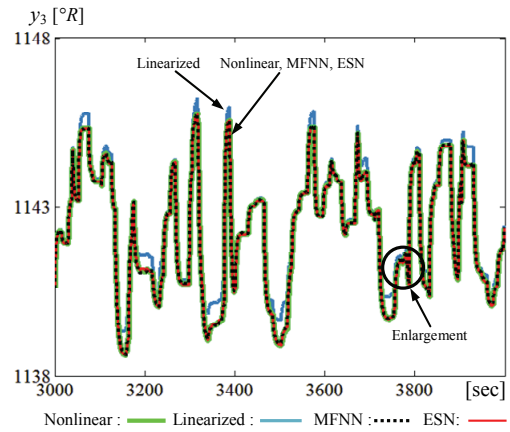*Un-Chul Moon, Jaewoo Lim and Kwang Y. Lee*

Fig. 9. Comparison of y_1 for validation data

Fig. 10. Comparison of y_2 for validation data
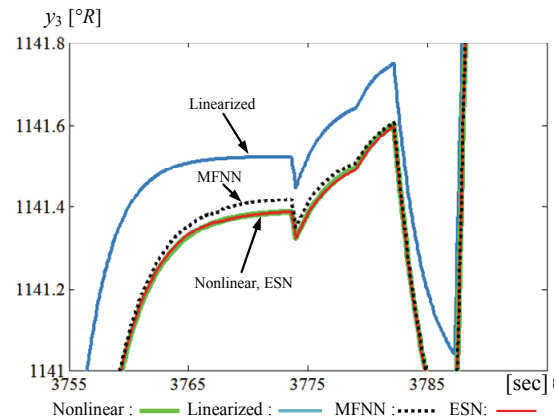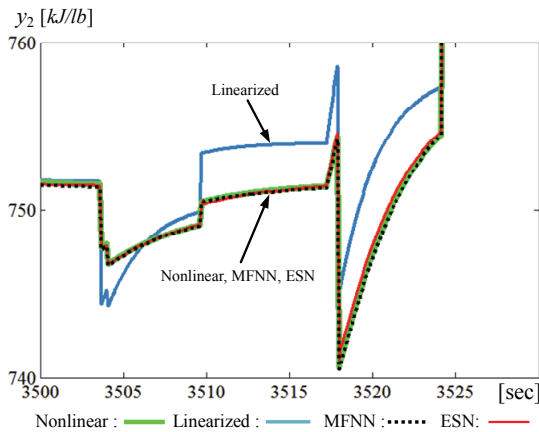

Fig. 11. An enlargement of Fig. 10

Fig. 10 shows the validation results for y_2, and Fig. 11 is an enlargement of Fig. 10. Since y_2 is a nonlinear output, the linear model shows some mismatch. This mismatch is expected from the analysis in Section 2.2. In this case, the MFNN and ESN show almost the same results as the nonlinear system. These validation results show that MFNN and ESN have the capability to describe the y_2 of the nonlinear water wall model properly. Fig. 12 shows the validation results of y_3, and Fig. 13 is an enlargement of

Fig. 12. Comparison of y_3 for validation data

Fig. 13. An enlargement of Fig. 12

Table 2. Comparison of MSE for three models

	Linear Model	MFNN Model	ESN Model
y_1	0.0000	4.8583e-13	2.3554e-26
y_2	1.5637e-04	4.1642e-06	1.1530e-06
y_3	2.1470e-03	1.4864e-05	5.3108e-06

Table 3. Comparison of the training time between MFNN and ESN

	MFNN Model	ESN Model
Training time [sec]	3853.6	3.5

Fig. 12. Since y_3 is a nonlinear output, the linear model shows some mismatch. The MFNN and ESN show almost the same results as the nonlinear system in the figure. These validation results show that MFNN and ESN have the capability to properly describe the entire dynamics of the nonlinear water wall model.

A quantitative comparison with the mean square error is listed in Table 2. Although both MFNN and ESN exhibit better results than the linear model, ESN shows slightly better results than the MFNN in this case. Table 3 shows the training time for the MFNN model and the ESN model. The simple and fast learning time of ESN is one of the

most important benefits over other neural network models.

5. Conclusion

In this paper, we present a comparison among linear model, multilayer feed-forward neural network (MFNN) model and echo state neural network (ESN) model for a practical water wall system in a 600 [MW] thermal power plant, which is a MIMO nonlinear system.

First, we present an analysis of the results of the linearization of a mathematical water wall model. The change in the pole and steady-state gain is presented as a function of the electric power output. As a result, the system shows quite severe nonlinearity in the design of a linear controller.

Second, we developed an MFNN model and an ESN model without using a mathematical water wall model. Though ESN shows slightly better performance, the results show that both neural network models can provide a satisfactory description of the nonlinear water wall model. Meanwhile, the ESN shows significantly faster training than the MFNN.

From the point of view of control system design, the change in the pole and steady-state gain in this paper can be applied. Though the use of ESN for control and analysis remains to be further developed in future research, we identify the ability of the ESN as a good neural network model for practical nonlinear MIMO identification problems.

Acknowledgements

References

[1] C. Liu, H. Wang, J. Ding, and C. Zhen, "An Overview of Modelling and Simulation of Thermal Power Plant," *Proc. of the 2011 International Conference on Advanced Systems*, pp. 86-91, Zhengzhou, China, 2011.

[2] L. Xueqin, L. Gang, and L. Shangqing, "The development of the boiler Water Wall Tube Inspection," *Third International Conference on Electric Utility Deregulation and Restructuring and Poewer Tech-*

nologies, pp. 2415-2420, Nanjing, April 2008.

[3] P. B. Usoro, "Modeling and Simulation of a Drum Boiler-Turbine Power Plant Under Emergency State Control," *Master Degree of Massachusetts Institute of Technology*, pp.138-139, May 1977.

[4] U.-C. Moon, and K. Y. Lee, "An Adaptive Dynamic Matrix Control with Fuzzy-Interpolated Step-Response Model for a Drum-Type Boiler-Turbine System," *IEEE Transactions on Energy Conversion*, Vol. 26, No. 2, pp. 393-401, June 2011.

[5] V. Kecman, *Learning and Soft Computing: Support Vector Machines*, *Neural Networks and Fuzzy Logic Models*, Piscataway, NJ: IEEE MIT Press, 2002.

[6] I. Snadburg, J. Lo, C. Fancourt, J. Principe, S. Katagiri, and S. Haykin*, Nonlinear Dynamical Systems: Feed-forward Neural Network Perspectives*, New York: Wiley, 2001.

[7] F. J. Lin, Y. C. Hung, and S. Y. Chen, "FPGA-Based Computed Force Control System Using Elman Neural Network for Linear Ultrasonic Motor," *IEEE Transactions on industrial electronics*, Vol. 56, No. 4, pp. 1238-1253, April 2009.

[8] Ku, C. C. and K. Y. Lee, "Diagonal Recurrent Neural Network for Dynamic Systems Control," *IEEE Transactions on Neural Networks,* Vol. 6, pp. 144-156, January 1995.

[9] A. F. Atiya and A. G. Parlos, "New results on recurrent network training: Unifying the algorithms and accelerating convergence," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 697-709, May 2000.

[10] H. Jaeger, "The echo sate approach to analyzing and training recurrent neural networks-with an Erratum note," Fraunhofer Institute for Autonomous Intelligent Systems (AIS), January 26, 2010.

[11] D. Prokhorov, "Echo state networks: Appeal and challenges," in *Proc. IEEE International Conference Neural Networks (IJCNN05)*, Montreal, PQ, Canada, July 31-August 4, 2005, Vol. 2, pp. 905-910.

[12] H. Jaeger, M. Lukosevicius, D. Popovici and U. Siewert, "Optimization and applications of echo state networks with leaky-integrator neurons," *Neural Networks*, Vol. 20, special issue, pp. 335-352, 2007.

[13] X. Lin, Z. Yang, and Y. Song, "Intelligent stock trading system based on improved technical analysis and Echo State Network," *Expert Systems with Applications*, Vol. 38, pp. 11347-11354, 2011.

[14] K. Ishii, T. van der Zant, V. Becanovic and P. Ploger, "Optimization of Parameter of Echo State Network and Its Application to Underwater Robot," *SICE Annual Conference in Sapporo*, pp. 2800-2805, August 2004.

[15] J. Dai, P. Zhang, J. Mazumdar, R. G. Harley and G. K. Venayagamoorthy "A Comparison of MLP, RNN and ESN in Determining Harmonic Contribution from Nonlinear Loads," *34th Annual Conference of IEEE*

on Industrial Electronics, pp. 3025-3032, November 2008.

[16] Y. Pan, and J. Wang, "Model Predictive Control of Unknown Nonlinear Dynamical Systems Based on Recurrent Neural Networks," *IEEE Transactions on Industrial Electronics*, Vol. 59, No. 8, pp. 3089-3101, August 2012.

[17] R. Johansson, *System Modeling and Identification*, Englewood Cliffs, NJ, USA: Prentice-Hall, 1992.

[18] S. Haykin, *Neural Networks*, Ottawa, On, Canada: Maxwell Macmillan, 1994.

[19] J. Mazumdar, and R. G. Harley, "Utilization of echo state networks for differentiating source and non-linear load harmonics in the utility network," *IEEE Transactions on power Electronics*, Vol. 23, No. 6, pp. 2738-2745, November 2008.

[20] G. Venayagamoorthy, "Online design of an echo state network based wide area monitor for a multimachine power system," *Neural Networks*, Vol. 20, No. 3, pp. 404-413, April 2007.

[21] D. Li, M. Han, and J. Wang, "Chaotic time series prediction based on a novel robust echo state network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 5, pp. 787-799, May 2012.

[22] H. Jaeger, "A tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and echo state network approach," GMD-Report 159, German National Research Center for Information Technology, 2002.

[23] B. Zhang, D. J. Miller, and Y. Wang, "Nonlinear system modeling with random matrices: Echo state networks revisited," *IEEE Transactions on Neural Networks and Learning systems*, Vol. 23, No. 1, pp. 175-182, January 2012.

[24] A. Millea, "Explorations in echo state networks," Department of Artificial Intelligence, Groningen University, MA, Netherlands, S2064235, June 2014.

Un-Chul Moon received his B.S., M.S. and Ph.D. degrees from Seoul National University, Korea, in 1991, 1993 and 1996, respectively, all in Electrical Engineering. From 2000, he was with the Woo-Seok University, Korea, and from 2002 with the Chung-Ang University, Korea, where he is currently an Associate Professor of Electrical Engineering. His current research interests are in the areas of power system analysis, computational intelligence and automation.

Jaewoo Lim received his B.S. degree in Electrical and Electronics Engineering from the Chung-Ang University, Seoul, Korea, in 2014. He is currently working toward the M.S degree in Chung-Ang University of Electrical and Electronics Engineering. His research interests include control and modeling of fossil power plant and power system analysis.

Kwang Y. Kee (F'01) received the B.S. degree in Electrical Engineering from Seoul National University, Seoul, Korea, in 1964, the M.S. degree in Electrical Engineering from North Dakota State University, Fargo, in 1968, and the Ph.D. degree in System Science from Michigan State University, East Lansing, in 1971. He has been on the faculties of Michigan State, Oregon State, University of Houston, Penn State, and Baylor University, where he is a Professor and Chair of Electrical and Computer Engineering. His interests are power system control, operation and planning and intelligent system techniques, and their application to power system and power plant control. Dr. Lee is an Editor of IEEE Transactions on Energy Conversion and former Associate Editor of IEEE Transactions on Neural Networks. Dr. Lee is a Fellow of the IEEE.