Estimation of Nuclear Reactor Vessel Water Level in Severe Accidents Using Cascaded Fuzzy Neural Networks

Dong Yeong Kim, Kwae Hwan Yoo, Geon Pil Choi, Man Gyun Na
Department of Nuclear Engineering
Chosun University
Gwangju, Republic of Korea
e-mail: doo891221@naver.com, yooqh@naver.com, zzangcgp7@naver.com, magyna@chosun.ac.kr

***Abstract*—The world's concern about nuclear reactor safety has increased considerably since the Fukushima accident. In case of most severe accidents, the nuclear reactor vessel water level cannot be measured. But, if the cascaded fuzzy neural network (CFNN) is used, under the event of severe accidents it might be possible to estimate the nuclear reactor vessel water level. The cascaded fuzzy neural network model can be used to estimate the nuclear reactor vessel water level value through the process of adding fuzzy neural networks (FNNs) repeatedly. The developed cascaded fuzzy neural network model is sufficiently accurate to be used to estimate the nuclear reactor vessel water level. Therefore, the developed cascaded fuzzy neural network model will be helpful for providing effective information for operators in severe accident situations.**

Keywords-Cascaded fuzzy neural network (CFNN); Fuzzy neural network (FNN); Nuclear reactor vessel water level.

I. INTRODUCTION

Recently, the world's concern about nuclear reactor safety has increased considerably since the Fukushima accident. The cause of these concerns and interest is because the operators do not quickly check the status of the plant in appropriate response to each situation.

The reactor vessel water level is essential information for confirming the cooling capability of the nuclear reactor core, to prevent the reactor core from melting down and to manage severe accidents effectively. In particular, decay heat is continuously generated in the reactor core after reactor shutdown. Therefore, it is important to estimate the reactor vessel water level to make provisions against severe accidents.

Many artificial intelligence techniques have been applied successfully to nuclear engineering areas, such as signal validation [1]–[3], plant diagnostics [4][5], event identification [6]–[9], etc. In this paper, a cascaded fuzzy neural network (CFNN) model is proposed to estimate the reactor vessel water level, which has a direct impact on the important times (time approaching the core exit temperature exceeding 1200°F, core uncovery time, reactor vessel failure time, etc.). The CFNN can be used to estimate the nuclear reactor vessel water level through the process of adding fuzzy neural networks (FNNs) repeatedly. To estimate the water level and the loss of coolant accident (LOCA) break size, other measured signals were used. The LOCA break size is not a measured variable. Instead, it is an estimated variable using the trend data for a short time early in the event preceding a severe accident. The LOCA classification algorithm for determining the LOCA position and LOCA break size estimation algorithm were explained in previous papers [10]-[12]. Because the LOCA break size could be estimated accurately, the LOCA break size was used as an input variable for estimating the reactor vessel water level. The obtained numerical simulation data was obtained and verified by simulating severe accident scenarios for the Optimized Power Reactor 1000 (OPR1000) using MAAP4 code [13].

Section II explains the methodology of CFNN including fuzzy inference system (FIS) and its training. Section III describes its application to estimating the water level in the reactor vessel.

II. CASCADED FUZZY NEURAL NETWORKS

A. Fuzzy inference system

The FIS uses the conditional rules that are comprised of an *if-then* rules of a pair of antecedent and consequent [14]. This study uses the Takagi-Sugeno-type FIS [15], which does not need the defuzzifier in the output terminal because its output is a real value. The Takagi-Sugeno-type FIS consists of three basic components without the defuzzifier block, differently from the Mamdani-type FIS shown in Figure 1.

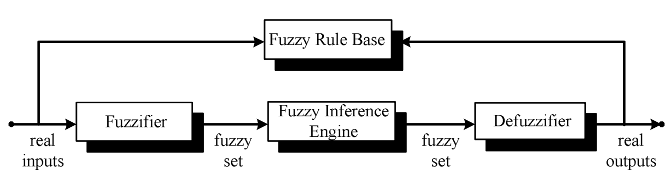

Figure 1. Fuzzy inference system (Mamdani-type FIS)

In the FIS, an arbitrary i^{th} fuzzy rule can be expressed as follows (first-order Takagi-Sugeno-type):

$$\begin{aligned}&\textit{If } x_1(k) \textit{ is } A_{i1}(k) \quad AND \cdots AND \ x_m(k) \textit{ is } A_{im}(k),\\ &\textit{then } \ y^i(k) \ \textit{ is } \ f^i(x_1(k),\cdots,x_m(k))\end{aligned} \tag{1}$$

where

$x_1, \cdots, x_m$: FIS input values

m = number of input variables

$A_{i1}(k), \cdots, A_{im}(k)$: fuzzy sets of the i^{th} fuzzy rule

y^i : output of the i^{th} fuzzy rule

$$f^i(x_1(k), \cdots, x_m(k)) = \sum_{j=1}^{m} q_{ij} x_j(k) + r_i \tag{2}$$

q_{ij} : weight of the j^{th} fuzzy input variable

r_i : bias of the i^{th} fuzzy rule

Because the function $f^i(x(k))$ is expressed as the first-order polynomial of input variables, FIS is called the first-order Takagi-Sugeno-type FIS in (2). The number of N input and output training data $\mathbf{z}^T(k) = (\mathbf{x}^T(k), y(k))$ (where $\mathbf{x}^T(k) = (x_1(k), x_2(k), \cdots, x_m(k)), k = 1, 2, \cdots, N$) are assumed to be available, and the input and output variables are normalized. In general, there is no special restriction on the shape of the membership functions. In this study, the symmetric Gaussian membership function is used to reduce the number of parameters to be optimized.

$$A_{ij}(x_j(k)) = e^{-(x_j(k) - c_{ij})^2 / 2s_{ij}^2} \tag{3}$$

The FIS output $\hat{y}(k)$ is calculated by weight-averaging the fuzzy rule outputs $y^i(k)$ as follows:

$$\begin{aligned} \hat{y}(k) &= \sum_{i=1}^{n} \overline{w}^i(k) y^i(k) = \sum_{i=1}^{n} \overline{w}^i(k) f^i(\mathbf{x}(k)) \\ &= \mathbf{w}^T(k)\mathbf{q} \end{aligned} \tag{4}$$

where

$$\overline{w}^i(k) = \frac{w^i(x(k))}{\sum_{i=1}^{n} w^i(x(k))} \tag{5}$$

$$w^i(k) = \prod_{j=1}^{m} A_{ij}(x_j(k)) \tag{6}$$

n : number of fuzzy rules

$$\mathbf{q} = [q_{11} \cdots q_{n1} \cdots\cdots q_{1m} \cdots q_{nm} r_1 \cdots r_n]^T$$

$$\begin{aligned} \mathbf{w}(k) = [&\overline{w}^i(k) x_1(k) \cdots \overline{w}^n(k) x_1(k) \cdots\cdots \\ &\overline{w}^1(k) x_m(k) \cdots \overline{w}^n(k) x_m(k)\ \overline{w}^1(k) \cdots \overline{w}^n(k)]^T \end{aligned}$$

The vector $\mathbf{q}$ is called a consequent parameter vector that has $(m+1)n$ dimensions, and the vector $\mathbf{w}(k)$ consists of input data and membership function values. The estimated output for a total of N input and output data pairs induced from (4) can be expressed as follows:

$$\hat{\mathbf{y}} = \mathbf{W}\mathbf{q} \tag{7}$$

where

$$\hat{\mathbf{y}} = [\hat{y}(1)\ \hat{y}(2) \cdots \hat{y}(N)]^T$$

$$\mathbf{W} = [\mathbf{w}(1)\ \mathbf{w}(2) \cdots \mathbf{w}(N)]^T$$

The matrix $\mathbf{W}$ has $N \times (m+1)n$ dimensions. Figure 2 describes the calculation structure of the FNN model [16].

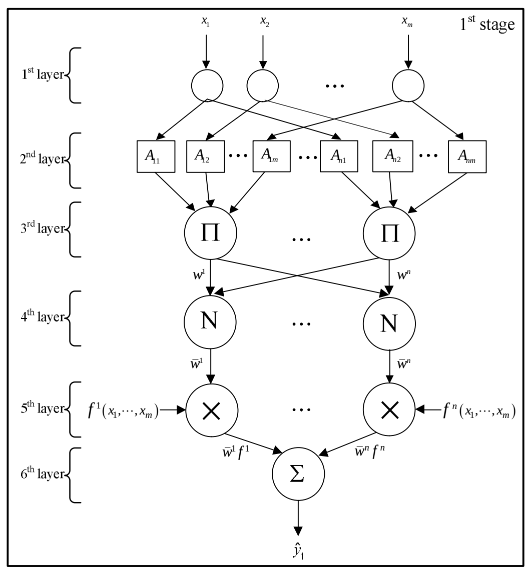

Figure 2. Fuzzy neural network (FNN)

B. *FIS training*

In this study, the FIS is optimized using the two combined methods of a genetic algorithm and a least squares method. The training data were used to develop the FNN model. The test data were used to verify the developed FNN model, and they are different from the training data set. The following fitness function for the genetic algorithm is proposed to minimize the maximum error and root mean square (RMS) error.

$$F = \exp(-\lambda_1 E_1 - \lambda_2 E_2) \tag{8}$$

 ISBN: 978-1-61208-437-4

where

$$E_1 = \sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}(y(k)-\hat{y}(k))^2}$$

$$E_2 = \max_k (y(k)-\hat{y}(k))$$

λ_1 : weighting value of RMS error

λ_2 : weighting value of maximum error

N_t : number of training data

The variable $y(k)$ is the actual output value, and $\hat{y}(k)$ is its value estimated using the FNN model. If the antecedent parameters are determined using a genetic algorithm through selection, crossover, and mutation, the resulting parameters appear similar to (7) as a first-order combination. Therefore, the consequent parameter $\mathbf{q}$ can be calculated easily using the least squares method. That is, the consequent parameter $\mathbf{q}$ is calculated to minimize an objective function. The objective function consists of the square error between the actual value $y(k)$ and its estimated value $\hat{y}(k)$, and it is expressed as follows:

$$\begin{aligned} J &= \sum_{k=1}^{N_t}(y(k)-\hat{y}(k))^2 = \sum_{k=1}^{N_t}(y(k)-w^T(k)q)^2 \\ &= \frac{1}{2}(\mathbf{y}_t-\hat{\mathbf{y}}_t)^2 \end{aligned} \tag{9}$$

where

$$\mathbf{y}_t = [y(1)\; y(2) \cdots y(N_t)]^T$$

$$\hat{\mathbf{y}}_t = [\hat{y}(1)\; \hat{y}(2) \cdots \hat{y}(N_t)]^T$$

A solution for minimizing the above objective function can be obtained using the following equation:

$$\mathbf{y}_t = \mathbf{W}_t\mathbf{q} \tag{10}$$

where

$$\mathbf{W}_t = [\mathbf{w}(1)\; \mathbf{w}(2) \cdots \mathbf{w}(N_t)]^T$$

The matrix $\mathbf{W}_t$ has $N_t \times (m+1)n$ dimensions in (10). The parameter vector $\mathbf{q}$ can be solved easily from the pseudo-inverse as follows:

$$\mathbf{q} = (\mathbf{W}_t^T\mathbf{W}_t)^{-1}\mathbf{W}_t^T\mathbf{y}_t \tag{11}$$

The parameter vector $\mathbf{q}$ can be calculated from a series of input and output data pairs and their membership function values because the matrix $\mathbf{W}_t$ consists of input data and membership function values.

C. Cascaded fuzzy neural networks

The foregoing FNN is composed of the fuzzy logic and neural network theory. Most of the existing FNN models have been proposed to implement different types of single-stage fuzzy reasoning mechanisms. However, single-stage fuzzy reasoning is only the most simple among a human being's various types of reasoning mechanisms. Syllogistic fuzzy reasoning, where the consequence of a rule in one reasoning stage is passed to the next stage as a fact, is essential to effectively build up a large scale system with high level intelligence [17]. Therefore, it is described by applying these techniques in this paper.

The CFNN model contains two or more inference stages where each stage corresponds to a single-stage FNN module. The architecture of the CFNN is shown in Figure 3.

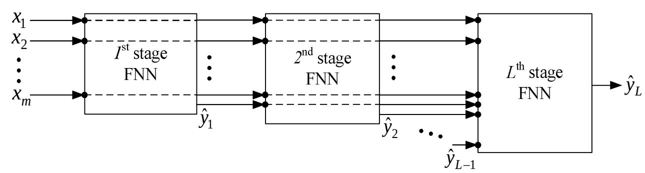

Figure 3. Cascaded fuzzy neural network (CFNN)

The CFNN can be used to estimate the target value through the process of adding FNN repeatedly. In CFNN method, the first stage FNN is the same as the FNN of Figure 2. The second stage FNN uses the initial input variables and the output variable of the first stage FNN as input variable. Therefore, this process is repeated L times to find the optimum value if over-fitting phenomena do not appear.

Similarly to (1), an arbitrary i^{th} rule of the CFNN can be expressed as (12):

$$\begin{aligned}
&\text{Stage 1} \begin{bmatrix} If\ x_1(k)\ is\ A_{i1}^1(k)\ AND \cdots AND\ x_m(k)\ is\ A_{im}^1(k), \\ then\ \hat{y}_1^i(k)\ is\ f_1^i(x_1(k),\cdots,x_m(k)) \end{bmatrix} \\
&\text{Stage 2} \begin{bmatrix} If\ x_1(k)\ is\ A_{i1}^2(k)\ AND \cdots AND\ x_m(k)\ is\ A_{im}^2(k) \\ \quad AND\ \hat{y}_1(k)\ is\ A_{i(m+1)}^2(k), \\ then\ \hat{y}_2^i(k)\ is\ f_2^i(x_1(k),\cdots,x_m(k),\hat{y}_1(k)) \end{bmatrix} \\
&\qquad\vdots \\
&\text{Stage } L \begin{bmatrix} If\ x_1(k)\ is\ A_{i1}^L(k)\ AND \cdots AND\ x_m(k)\ is\ A_{im}^L(k), \\ \quad AND\ \hat{y}_1(k)\ is\ A_{i(m+1)}^L(k)\ AND \cdots AND\ \hat{y}_{(L-1)}(k)\ is\ A_{i(m+L-1)}^L(k), \\ then\ \hat{y}_L^i(k)\ is\ f_L^i(x_1(k),\cdots,x_m(k),\hat{y}_1(k),\cdots,\hat{y}_{(L-1)}(k)) \end{bmatrix}
\end{aligned} \tag{12}$$

where L is the stage number of CFNN. The CFNN model is trained sequentially at each FNN module in the same way as explained in subsections II. *B* and II. *C*.

III. Application to Estimating the Nuclear Reactor Vessel Water Level

The proposed CFNN model was applied to estimating the water level in the reactor vessel. The training and test data of the proposed model was acquired by simulating the severe accident scenarios using the MAAP4 code concerning the OPR1000 nuclear power plant.

The simulation data is divided into the break position and break size of LOCA. The break position was divided into hot-leg LOCA, cold-leg LOCA and SGTR, and the break size was divided into a total of 200 steps.

The LOCA position was identified completely and the LOCA break size was estimated accurately in previous studies [10]-[12], with an approximately 1% error level. Therefore, the LOCA break size signal, which is an input signal to the FNN model, is assumed to be estimated from the algorithms of previous studies. Through the simulations, a total of 600 cases of severe accident scenarios are obtained. This data is composed of 200 pieces of hot-leg LOCA, 200 pieces of cold-leg LOCA and 200 pieces of SGTR.

The test data were different from the data that were used to develop the CFNN model, and consisted of the time that elapsed after reactor shutdown, the estimated LOCA break size, and the containment pressure. At this study, 300 data points in each of the LOCA break positions, namely, hot-leg and cold-leg LOCA, and SGTR, were selected as test data points.

The parameter values that are concerned with the genetic algorithm and the FIS are as follow: the crossover 100%, the mutation probability is 5%, and the population size is 20.

Table I shows the performance results that were obtained with the CFNN model for the break positions of hot-leg, cold-leg and SGTR, respectively.

TABLE I. Performance of the CFNN Model

RMS error(m)	All break sizes		
	Hot-leg LOCA	**Cold-leg LOCA**	**SGTR**
2 fuzzy rules	0.1721	0.2130	0.3399
3 fuzzy rules	0.2280	0.1895	0.3351
5 fuzzy rules	0.2255	0.7045	0.3233
7 fuzzy rules	0.1380	13.6493	0.3261

For the test data of the hot-leg LOCA, the RMS errors were approximately 0.17m, 0.23m, 0.23m, and 0.14m for the CFNN model with 2, 3, 5, and 7 fuzzy rules, respectively. And the RMS errors were approximately 0.21m, 0.19m, 0.70m, and 13.65m for the test data of the cold-leg LOCA and 0.34m, 0.34m, 0.32m, and 0.33m for the test data of the SGTR for the CFNN model with 2, 3, 5, and 7 fuzzy rules, respectively. Therefore, the CFNN model with 7 fuzzy rules proved to be the most accurate for estimating the nuclear reactor vessel water level in hot-leg LOCA and the CFNN model for cold-leg LOCA is 3 fuzzy rules, while the CFNN model with 5 fuzzy rules was shown to be the most accurate for estimating the nuclear reactor vessel water level in SGTR.

The CFNN models have been shown to be capable of accurately estimating the nuclear reactor vessel water level in case of a severe accident.

IV. Conclusion

In this study, a CFNN model was developed to estimate the nuclear reactor vessel water level in severe accident. The developed CFNN model is verified based on the simulation data of OPR1000 using MAAP4 code. The simulation results show that the performance of the developed CFNN model is quite accurate with about approximately 2% error. The developed CFNN model will be helpful for providing effective information for operators in severe accident situations.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2012M2B2B1055611).

References

[1] J. W. Hines, D. J. Wrest, and R. E. Uhrig, “Signal validation using an adaptive neural fuzzy inference system,” Nucl. Technol., vol. 119, no. 2, pp. 181-193, Aug. 1997.

[2] M. G. Na, “A neuro-fuzzy inference system for sensor failure detection using wavelet denoising, PCA and SPRT,” J. Korean Nucl. Soc., vol. 33, no. 5, pp. 483-497, Oct. 2001.

[3] J. Garvey, D. Garvey, R. Seibert, and J. W. Hines, “Validation of on-line monitoring techniques to nuclear plant data,” Nucl. Eng. Tech., vol. 39, no. 2, pp. 149-158, Apr. 2007.

[4] E. B. Bartlett and R. E. Uhrig, “Nuclear power plant diagnostics using an artificial neural network,” Nucl. Technol., vol. 97, pp. 272-281, March 1992.

[5] M. Marseguerra and E. Zio, “Fault diagnosis via neural networks: The Boltzmann machine,” Nucl. Sci. Eng., vol. 117, no. 3, pp. 194-200, July 1994.

[6] Y. Gyu No, J. H. Kim, M. G. Na, D. H. Lim, and K.-I. Ahn, “Monitoring Severe Accidents Using AI Techniques,” Nucl. Eng. Technol., vol. 44, no. 4, pp. 393-404, May 2012.

[7] M. G. Na, et al., “Prediction of major transient scenarios for severe accidents of nuclear power plants,” IEEE Trans. Nucl. Sci., vol. 51, no. 2, pp. 313-321, April 2004.

[8] S. W. Cheon and S. H. Chang, “Application of neural networks to a connectionist expert system for transient identification in nuclear power plants,” Nucl. Technol., vol. 102, no. 2, pp. 177-191, May 1993.

[9] Y. Bartal, J. Lin, and R. E. Uhrig, “Nuclear power plant transient diagnostics using artificial neural networks that allow “don’t-know” classifications,” Nucl. Technol., vol. 110, no. 3, pp. 436-449, June 1995.

[10] S. H. Lee, Y. G. No, M. G. Na, K.-I. Ahn, and S.-Y. Park, “Diagnostics of Loss of Coolant Accidents Using SVC and GMDH Models,” IEEE Trans. Nucl. Sci., vol. 58, no. 1, pp. 267-276, Feb. 2011.

[11] M. G. Na, W. S. Park, and D. H. Lim, “Detection and Diagnostics of Loss of Coolant Accidents Using Support Vector Machines,” IEEE Trans. Nucl. Sci., vol. 55, no. 1, pp. 628-636, Feb. 2008.

[12] M. G. Na, S. H. Shin, D. W. Jung, S. P. Kim, J. H. Jeong, and B. C. Lee, “Estimation of Break Location and Size for Loss of Coolant Accidents Using Neural Networks,” Nucl. Eng. Des., vol. 232, no. 3, pp. 289-300, Aug. 2004.

[13] R. E. Henry, et al., MAAP4 – Modular Accident Analysis Program for LWR Power Plants, User’s Manual, Fauske and Associates, Inc., vol. 1, 2, 3, and 4, 1990.

[14] E. H. Mamdani and S. Assilian, “An experiment in linguistic synthesis with a fuzzy logic controller,” Int. J. Man-Machine Studies, vol. 7, pp. 1-13, 1975

[15] T. Takagi and M. Sugeno, “Fuzzy identification of systems and its applications to modeling and control,” IEEE Trans. Systems, Man, Cybern., vol. SMC-1, no. 1, pp. 116-132, Jan./Feb. 1985.

[16] D. Y. Kim, K. H. Yoo, J. H. Kim, M. G. Na, S. Hur, and C-H. Kim, “Prediction of Leak Flow Rate Using Fuzzy Neural Networks in Severe Post-LOCA Circumstances,” IEEE Trans. Nucl. Sci., Vol. 61, No. 6, pp. 3644-3652, Dec. 2014.

[17] J. C. Duan and F .L. Chung, “Cascaded Fuzzy Neural Network Model Based on Syllogistic Fuzzy Reasoning,” IEEE Trans. Fuzzy Systems, vol. 9, no. 2, pp.293-306, Apr. 2001.