Proceedings of the Korean Nuclear Society Spring Meeting
Pohang, Korea, May 1999

A Genetic Neuro-Fuzzy Logic for DNB Protection

Man Gyun Na

Chosun University
Department of Nuclear Engineering
375 Seosuk-dong, Dong-gu, Kwangju 501-759, Korea

Abstract

A neurofuzzy method is used to estimate the DNB protection limit using the measured average temperature and pressure of a reactor core. The neurofuzzy system parameters are optimized by two learning methods. A genetic algorithm is used to optimize the antecedent parameters of the neurofuzzy inference system and a least-squares algorithm to solve the consequent parameters. Two neurofuzzy inference systems are used according to the pressure and temperature regions. The proposed method is applied to the Yonggwang 3&4 nuclear power plant and the proposed method has 5.84 percent larger thermal margin than the conventional Westinghouse $OT\Delta T$ trip logic. This simple algorithm can provide a good information for the nuclear power plant operation and diagnosis by estimating the DNB protection limit each time step.

1. Introduction

The DNB (Departure from Nucleate Boiling) correlation provides the expected value of fuel rod surface heat flux that will cause DNB for various coolant conditions and flow geometry. The ratio of the expected DNB heat flux to the actual fuel rod heat flux at a particular time during an incident is called the DNBR (DNB Ratio) at that time. A correlation limit DNBR is established based on the variance of the correlation such that there is a 95 percent probability at a 95 percent confidence level that DNB will not occur when the calculated DNBR is at the correlation limit DNBR. The conservative design method that the calculated DNBR is greater than the correlation limit DNBR on the limiting power rod is established by considering all parameters at fixed conservative values. The variable value design method used in this work assures that the DNBR on the limiting power rod is greater than the correlation limit DNBR by statistically combining the effects of uncertainties of the input parameters. Therefore, the design limit DNBR applicable to all Condition I and II events is determined by utilizing the DNBR sensitivities and variances in three input parameter categories: plant operating parameters, nuclear and thermal parameters and fabrication parameters [1].

The protection system of the conventional pressurized water reactors designed by Westinghouse is an analog system. However, the Korea Standard Nuclear Power Plant (KSNPP) and the recently designed nuclear reactors employ a digital protection system. The ABBCE-type nuclear power plants employ the Core Protection Calculator System (CPCS) which continuously calculates DNBR and Local Power Density (LPD) in order to assure that the specified acceptable fuel design limits such as DNB and fuel centerline melt are not exceeded during anticipated operational occurrences. The CPCS has approximately 6,000 constants and the CPCS is designed by deciding the CPCS constants [2]. This large number of constants makes the software V&V (Verification and Validation) more difficult.

Since the conventional Westinghouse DNB protection logic is implemented on analog circuits, the logic must be very simple. The Westinghouse $OT\Delta T$ protection logic heavily restricts the operation region by applying the same logic for a full range of pressure in order to maintain its simplicity. However, if the DNB protection logic is implemented in a digital processor, a little complexity may be allowed to increase the thermal (or operation) margin.

The objective of this work is to estimate the DNB protection limit according to operating conditions by using a neurofuzzy method and to compare the thermal margin by this method and that by the conventional method. The effect of the neutron flux difference between upper and lower half cores will be neglected. In other words, this effect is assumed to be the same in the two methods.

Fuzzy system parameters such as membership functions and the connectives between layers in a neural network are tuned by two learning methods to minimize the errors between the target values and the trained values. A genetic algorithm is applied to optimize the membership function parameters and a least-squares algorithm to solve the connectives. Since DNB data have two different characteristics according to the pressure and temperature region, two neurofuzzy inference systems are used according to the pressure and temperature regions.

The proposed method was applied to the Yonggwang 3&4 nuclear power plant.

2. Design of a Genetic Neurofuzzy Inference System

A fuzzy inference system consists of situation and action pairs where conditional rules in *if/then* statements are generally used. Adapting fuzzy systems would be the desirable objective. Such neuronal improvements of fuzzy systems as well as the fuzzification of neural network systems aim at exploiting the complementary nature of the two approaches; the fuzzy and neural network systems. Their composite is usually called as a neurofuzzy system.

In a fuzzy inference system, the i-th rule can be described using the first-order Sugeno-Takagi type [3] as follows.

$$\text{If } x_1 \text{ is } A_{i1} \ AND \cdots AND \ x_L \text{ is } A_{iL}, \text{ then } y_i \text{ is } f_i(x_1, \cdots, x_L), \tag{1}$$

where

$x_1, \cdots, x_L$ = input variables to the neurofuzzy inference system,

$A_{i1}, \cdots, A_{iL}$ = antecedent membership function of each input variable for the i-th rule($i = 1, 2, ..., n$),

y_i = output of the i-th rule,

$$f_i(x_1, \cdots, x_L) = \sum_{j=1}^{L} q_{ij} x_j + r_i, \tag{2}$$

q_{ij} = weighting value of the j -th input onto the i -th output,

r_i = bias of the i -th output.

n is the number of rules and L the number of input variables. There is no restriction on the shape of a membership function. In this work, for small number of the parameters that are tuned, the following symmetric Gaussian membership function is used.

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

where c_{ij} denotes a center position of a peak of a membership function for the i -th rule and the j -th input, and s_{ij} its sharpness.

The output of a neurofuzzy inference system with n rules is obtained by weighting the real values of consequent part for all rules with the corresponding membership grade. The output is obtained as follows.

$$y = \sum_{i=1}^{n} \overline{w}_i f_i , \tag{4}$$

where

$$\overline{w}_i = \frac{w_i}{\sum_{i=1}^{n} w_i} , \tag{5}$$

$$w_i = \prod_{j=1}^{L} A_{ij}(x_j) . \tag{6}$$

Equation (4) can be expressed as follows:

$$\begin{aligned} y = {} & \overline{w}_1 (x_1 q_{11} + \cdots + x_L q_{1L} + r_1) + \overline{w}_2 (x_1 q_{21} + \cdots + x_L q_{2L} + r_2) + \cdots + \\ & \overline{w}_n (x_1 q_{n1} + \cdots + x_L q_{nL} + r_n). \end{aligned} \tag{7}$$

The output can be rewritten as the following vector form:

$$y = \mathbf{w}^T \mathbf{q} , \tag{8}$$

where the vectors $\mathbf{w}$ and $\mathbf{q}$ are defined as

$$\mathbf{w} = [\overline{w}_1 x_1 \cdots \overline{w}_n x_1 \cdots \cdots \overline{w}_1 x_L \cdots \overline{w}_n x_L \;\; \overline{w}_1 \cdots \overline{w}_n]^T , \tag{9}$$

$$\mathbf{q} = [q_{11} \cdots q_{n1} \cdots \cdots q_{1L} \cdots q_{nL} \;\; q_{10} \cdots q_{n0}]^T . \tag{10}$$

The membership value for rule i , w_i , means a compatibility grade between antecedent parts of a rule. The multiplicative weight in Eq. (6) is preferred over the minimum weight because of its smoothness properties. The neurofuzzy inference system described above is shown in Fig. 1.

Fuzzy system parameters such as membership functions and the connectives between layers in a neurofuzzy inference system must be optimized for good performance. This is accomplished by adapting the antecedent parameters (membership function parameters) and consequent parameters (the polynomial coefficients of the consequent part) so that a specified objective function is minimized. The adaptation methods of most neurofuzzy systems rely on the gradient-descent optimization. However, in this work, two learning methods were combined to optimize the antecedent and consequent parameters because the method shows better performance than any

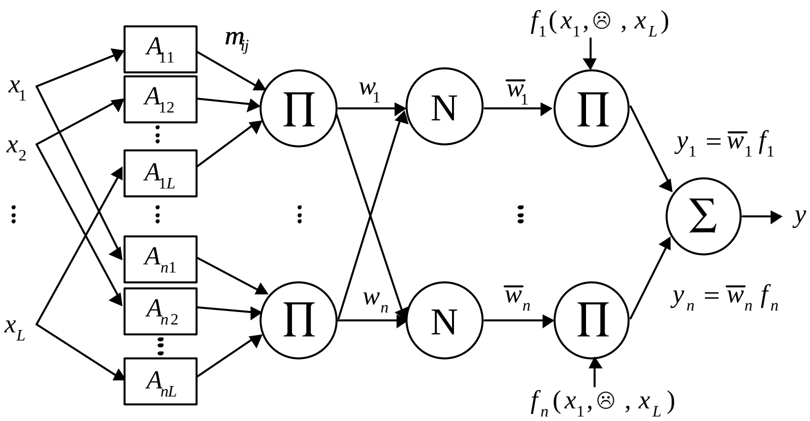

Fig. 1. Fuzzy inference system.

other method. The methods are a genetic algorithm and a least-squares method. The genetic algorithm is used to optimize the antecedent parameters c_{ij} and s_{ij}, and the least-squares algorithm to solve the consequent parameters q_{ij} and r_i.

2.1 Antecedent Parameter Optimization

Genetic algorithms for optimization were formally introduced in the 1970s by John Holland [4]. The continuing performance improvements of computational systems have made them attractive for some types of optimization. Many optimization methods move from a single point in the decision space to the next using some transition rule to determine the next point. This point-to-point method is dangerous because it is a perfect prescription for locating false peaks in many peaked search spaces. By contrast, genetic algorithms work from a rich database of points simultaneously climbing many peaks in parallel. Thus, the probability of finding a false peak is reduced over methods that go point to point. Therefore, genetic algorithms are less susceptible to getting stuck at local optima than conventional search methods.

Many search techniques require much auxiliary information such as derivatives in order to work properly. By contrast, genetic algorithms have no need for all this auxiliary information. Genetic algorithms use random choice as a tool to guide a search toward regions of the search space with likely improvement [5]. Despite of these advantages, however, genetic algorithms tend to be computationally expensive.

In genetic algorithms, the term *chromosome* typically refers to a candidate solution to a problem, often encoded as a bit string. Each chromosome can be thought of as a point in the search space of candidate solutions. The genetic algorithms process populations of chromosomes, successively replacing one such population with another. The genetic algorithms require a fitness function that assigns a score to each chromosome in the current population. The fitness of a chromosome depends on how well that chromosome solves the problem at hand [6].

After an initial population of chromosomes is randomly generated, the typical genetic algorithm evolves the population through the three operators; selection, crossover and mutation operators. The selection operator selects individuals (chromosomes) in the population for reproduction. The goodness of each individual depends on its fitness. The fitter the chromosome, the more times it is likely to be selected to be reproduced. After two individuals are chosen from the population using the selection operator, the crossover operator randomly chooses a crossover site along the bit strings and exchanges the subsequences before and after that crossover site between the two individuals to create two offspring. The two new offspring created from this mating are put into the next generation of the population. By recombining portions of good individuals, this process is likely to

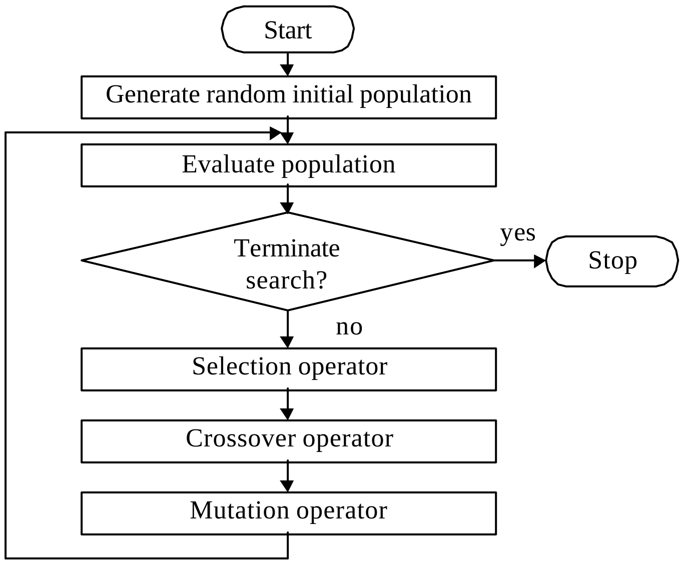

Fig. 2. Diagram of genetic algorithms.

create even better individuals. With some low probability, a portion of the new individuals will have some of their bits flipped. Mutation can occur at each bit position in a string with some probability, usually very small. Its purpose is to maintain diversity within the population and inhibit premature convergence.

Most genetic algorithms follow the procedures in the Fig. 2 as explained above.

The number of the antecedent parameters in the neurofuzzy inference system is $2 \times L \times n$ in case that a membership function has two parameters. L is the number of input variables and n is the number of the rules. To use a genetic algorithm, a solution to a given problem must be represented as a chromosome. The genetic algorithm then creates a population of solutions and applies genetic operators such as selection, crossover and mutation to evolve the solutions in order to find the best one. The three most important aspects of using genetic algorithms are (1) definition of the objective function, (2) definition and implementation of the genetic representation, and (3) definition and implementation of the genetic operators.

A genetic algorithm uses a cost function that evaluates the extent to which each individual is suitable for the given objectives such as maximum error together with small overall error. The fitness of an individual (chromosome) is calculated by means of the energy of the individual. Each chromosome contains the antecedent parameters c_{ij} and s_{ij}. The chromosome that has lower energy has higher fitness. The energy functions are defined by the following two equations.

$$E_1 = \sum_{k=1}^{N} \left(y_{rk} - y_k \right)^2 , \tag{11}$$

$$E_2 = \max\left\{ \left| y_{r1} - y_1 \right|, \left| y_{r2} - y_2 \right|, \cdots, \left| y_{rN} - y_N \right| \right\}, \tag{12}$$

where

N = number of data pairs,

y_{rk} = target output for the k-th input data $\left(x_1, \cdots, x_L \right)$,

y_k = output calculated from a neurofuzzy inference system for the same input data.

E_1 and E_2 are overall sum of squared errors and maximum absolute error, respectively. The fitness function is given as follows.

$$F = \exp\left(-\boldsymbol{a} E_1 - \boldsymbol{b} E_2 \right), \tag{13}$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ are weighting coefficients.

To increase the efficiency of the conventional genetic algorithm, three methods are applied [7]. At first, the proposed genetic algorithm has initial coarse tuning characteristics by initially representing each parameter in a chromosome by a small bit number. If the parameters in a chromosome are represented by big bit numbers, the genetic algorithm can find the accurate optimal points in a limit of resolution but needs much more time to reach a convergence point. Therefore, it is necessary to represent it by a big bit number as many chromosomes (solution) gradually approach the optimal points. By this method, the genetic algorithm has initial coarse tuning and final fine tuning characteristics. The crossover site is selected by two ways. The first is that the crossover site is selected randomly in a chromosome. The second is that the crossover site is selected between only parameters in a chromosome. This method slows a premature convergence without reaching optimal solutions and speeds up a final convergence. Also, a portion of the population of chromosomes with higher fitness in a priori generation is added to a new generation. And then, the same portion of the population of chromosomes with lower fitness in the total new generation is removed. This is to inhibit final drifting without convergence.

2.2 Consequent Parameter Optimization

The back-propagation algorithm was developed to train the fuzzy logic system so as to match desired output-actual output pairs. Because the fuzzy logic system is nonlinear in its adjustable parameters, the back-propagation algorithm implements a nonlinear gradient optimization procedure and can be trapped at a local minimum and converges slowly. If we fix some parameters of the fuzzy logic system, the resulting fuzzy system is equivalent to a series expansion of some basis functions. This basis function expansion is linear in its adjustable parameters. Therefore, we can use the least-squares method to determine the remaining parameters. For example, if we fix the membership function parameters in the first-order Takagi-Sugeno fuzzy model, the inference system output is written by Eq. (8). In case that there exist N data pairs $(x_1, x_2, \cdots x_L, y)$, the output y and inputs $x_1, x_2, \cdots x_L$ are N - dimensional column vectors, the consequent parameters are chosen such that the data satisfy the following equation:

$$\mathbf{y} = \mathbf{W}^T \mathbf{q}, \tag{14}$$

where $\mathbf{y}$ is the output data and the matrix $\mathbf{W}$ from Eq. (9) includes the input data defined as, respectively

$$\mathbf{y} = \left[y^1 \ y^2 \cdots \ y^N \right]^T,$$

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

N is the number of the input-output data pairs. The controller outputs are represented by $N \times (L+1)n$ - dimensional matrix with N rows equal to the number of data pairs and $(L+1)n$ columns. In order to solve the parameter vector $\mathbf{q}$ in Eq. (14), the matrix $\mathbf{W}$ should be invertible but is not usually a square matrix. However, the vector can be solved using the pseudoinverse as follows:

$$\mathbf{q} = \left(\mathbf{W}\mathbf{W}^T\right)^{-1} \mathbf{W}\mathbf{y}. \tag{15}$$

The least-squares method is a one-pass regression procedure and is therefore much faster than the back-propagation algorithm.

2.3 DNB Protection

Using the optimized parameters, the ΔT (the temperature difference between the hot leg and the cold leg) value is calculated where DNB may take place at a measured pressure and a measured average temperature. The ΔT protection limit is established based on the measurement errors and the variance of the estimated value so that there is a 95 percent probability at a 95 percent confidence level that DNB will not occur when the calculated ΔT is at the ΔT protection limit. Therefore, the ΔT protection limit is defined as follows:

$$\Delta T_{sp} = \Delta T_{estimate} - \boldsymbol{e}\Delta T_o - 1.645\,\boldsymbol{s}, \tag{16}$$

where

$\Delta T_{estimate}$ = ΔT calculated by the proposed algorithm,

ΔT_o = rated ΔT,

$\boldsymbol{e}$ = measurement uncertainty (refer to Table 1),

$\boldsymbol{s}$ = standard deviation of difference between the actual values and the estimated values.

3. Application to the Yonggwang 3&4 Nuclear Power Plant

The design limit DNBR was assumed to be 1.493 (refer to [2][8]). Also, in order to generate DNB data, the same assumptions that had been used in obtaining the Westinghouse DNB protection limit were applied. The major assumptions used in calculation are as follows [9]:

1) The axial power distribution is a 1.55 chopped cosine shape.

2) The nuclear enthalpy rise hot channel factor $\left(F_{\Delta H}^{N}\right)$ is a design hot channel factor $F_{d\Delta H}^{N}$ for 100 percent rated or greater power levels, and for power levels less than the rated power $F_{\Delta H}^{N}$ is given by

$$F_{\Delta H}^{N}(P) = F_{d\Delta H}^{N}\left[1+0.3(1-P)\right], \tag{17}$$

where P = power level.

3) The coolant flow rate is the design value that is usually about 5% less than the best estimate flow.

4) The bypass flow is excluded from the available core flow.

5) The coolant flow to the hottest coolant channel is reduced by 5 percent.

Except for the above parameters, the other DNB input parameters were considered to be nominal values. As the core power and inlet temperature vary at a given pressure, the vessel average hot-leg temperature is calculated using the COBRA code when the minimum DNBR of the limiting power rod is equal to the design limit DNBR. The DNB data calculated from the COBRA code are given in Fig. 3. Figure 3 shows two different characteristics as denoted by symbols 'x' and '+'. Therefore, two neurofuzzy inference systems are used according to the pressure and temperature regions separated by the straight line as shown in Fig. 4. About a half of the DNB data are used to train the neurofuzzy inference system and the remaining data are used to verify the inference system.

The number of the inputs into each neurofuzzy inference system is two and the inputs to the inference system are the pressure x_1 and average temperature x_2. Also, the target output y_r is ΔT.

The tuning parameters that are tuned by the genetic algorithm are the center value c_{ij} and standard deviation s_{ij} of a peak of the Gaussian function. Their initial values were randomly represented by the chromosomes of the population. The number of rules in an inference system was selected to be 2. The initial values for this genetic algorithm are as follows;

Population size = 30

Crossover probability = 100%

Mutation probability = 3%

Maximum generation = 200

Constants for the fitness function; $a = 1$, $b = 10$.

After the inference systems are trained for 400 generations, the results are shown in Figs. 5-10. The consequent parameters are calculated by Eq. (15) for a population of chromosomes every generation. The finally trained membership functions are shown in Figs. 5 and 6.

Figure 7 shows the actual and estimated values. The ΔT values where DNB occurs at each pressure and average temperature are estimated accurately. Figure 8 shows the total squared error and maximum error between the estimated ΔT and the actual ΔT. The total squared error and maximum error decrease gradually as the generation increases. Figure 9 shows the fitness functions. The fitness function increases gradually. Figure 10 shows the distribution of the errors between the estimated ΔT and target ΔT. Its distribution is similar to the Gaussian distribution. Based on this distribution, the standard deviation s is $0.091164\,^{o}F$. In order to have more conservative feature, the measurement uncertainty and the $1.645s$ are subtracted from the estimated ΔT to obtain the DNB protection limit. Figure 11 shows the case that the proposed algorithm was applied to data that had not been used for the training. From the figures 5-11, although the algorithm was applied to the arbitrary measured pressure and average temperature, it is known that it gives an accurate DNB protection limit.

A steady-state thermal margin was compared between the Westinghouse DNB protection system and the proposed one. The thermal margin may be defined as $\Delta T_{sp} / \Delta T_0$ at nominal cold leg temperature and RCS pressure. The nominal cold leg temperature and RCS pressure are $564.5^o F$ and $2250\,psia$, respectively. The rated ΔT, ΔT_0 is given as $56.5^o F$ and the ΔT protection limit calculated by the proposed algorithm is $69.3308^o F$ at the nominal cold leg temperature and RCS pressure. Therefore, the thermal margin of the proposed algorithm is 122.62%.

For a 1.55 chopped cosine shape at steady-state condition, the Westinghouse $OT\Delta T$ trip setpoint is determined from the following equation [8]:

$$\Delta T_{sp} = \Delta T_o \left[K_1 - K_2\left(T_{avg} - T_{avgo}\right) + K_3\left(P - P_o\right)\right], \tag{18}$$

where

ΔT_o = indicated ΔT at nominal plant conditions,

ΔT_{avg} = measured average coolant temperature,

ΔT_{avgo} = reference average coolant temperature at nominal plant conditions of rated power,

P = measured RCS pressure,

P_o = reference RCS pressure at nominal plant conditions of rated power,

The thermal margin of the Westinghouse DNB protection system may be written as [8]

$$\frac{\Delta T_{sp}}{\Delta T_o} = \frac{K_1 + K_2 \cdot \dfrac{\Delta T_o}{2}}{1 + K_2 \cdot \dfrac{\Delta T_o}{2}}, \tag{19}$$

where

$K_1 = 1.2382$ (from ref. [8] and Table 1)

$K_2 = 0.014846$ (from ref. [8])

$T_o = 56.5^o F$.

The conventional $OT\Delta T$ trip logic has the thermal margin 116.78 percent. The proposed method has 5.84% larger thermal margin than the conventional $OT\Delta T$ trip logic.

4. Conclusions

A neurofuzzy inference method was applied to estimate the DNB protection limit using the measured average temperature and pressure. Fuzzy system parameters such as the membership functions and the connectives between layers in a neural network are optimized by two learning methods. The learning methods are a genetic algorithm to optimize the antecedent parameters and a least-squares algorithm to solve the consequent parameters.

The network was trained by using DNB data of the Yonggwang 3 and 4 units. The DNB data are the inlet and outlet temperatures where the minimum DNBR of the limiting power rod is equal to the design limit DNBR at a given pressure. The inputs to the neurofuzzy inference system are the average temperature and pressure of the reactor core and the output is the temperature difference ΔT between inlet and outlet. Since DNB data have two different characteristics according to the pressure and average temperature, two neurofuzzy inference systems are used. The ΔT, which induces DNB at a given average temperature and pressure, is estimated from this proposed algorithm. The measurement error and the uncertainty of the estimation algorithm are subtracted from the estimated ΔT in order to establish the setpoint ΔT so that this algorithm has some conservative features. Even though the rule number of this algorithm is small, the estimate is accurate. The proposed algorithm has 5.84 percent larger thermal margin than the conventional $OT\Delta T$ logic.

It is recommended to accomplish more realistic and exact DNB protection limit by adding the coolant flow rate and axial power shape to the input of the neurofuzzy inference system using the DNB data on the coolant flow rate and axial power shape.

References

1. H. Chelemer, L.H. Boman and D.R. Sharp, "Improved Thermal Design Procedure," WCAP-8567 (1975).
2. G.S. Auh, D.H Hwang, and S.H. Kim, "A Steady-State Margin Comparison Between Analog and Digital Protection Systems," *J. KNS*, Vol. 22, No. 1, pp. 45-57 (1990).
3. T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Trans. System, Man, Cybern.*, Vol. 1, pp. 116-132 (1985).
4. J. H. Holland, *Adaptation in Natural and Artificial Systems*, Univ. Michigan Press, Ann Arbor, MI, 1975.
5. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, Massachusetts, 1989.
6. M. Mitchell, *An Introduction to Genetic Algorithms*, The MIT Press, Cambridge, Massachusetts, 1996.
7. Man Gyun Na, "Design of a Genetic Fuzzy Controller for the Nuclear Steam Generator Water Level Control," *IEEE Trans. Nucl. Sci.*, Vol. 45, No. 4, pp. 2261-2271 (1998).
8. Man Gyun Na, "On-line Estimation of DNB Protection limit via a fuzzy Neural Network," *J. KNS*, Vol. 30, No. 3, pp. 222-234 (1998).
9. K.I. Han, " $OP\Delta T$ and $OT\Delta T$ Trip Setpoint Generation Methodology," *J. KNS*, Vol. 16, No. 2, pp. 106-115 (1984).

Table 1. Measurement errors of DNBR calculations [2].

<table>
<tr><th></th><th colspan="2">Parameters</th><th>Range R</th><th>Variance ($\sigma^2 = R^2/12$)</th></tr>
<tr><td rowspan="4">Errors</td><td rowspan="3">Calibration</td><td>Calorimetric</td><td>4.0%</td><td>1.3</td></tr>
<tr><td>$T_{avg}(\pm 2^{o} F)$</td><td>4.9%</td><td>2.01</td></tr>
<tr><td>Pressure ($\pm 8 psi$)</td><td>1.5%</td><td>0.19</td></tr>
<tr><td colspan="2">Signal linearity, reproducibility, and bistable error</td><td>10.73%</td><td>9.6</td></tr>
<tr><td>Total Variance</td><td colspan="4">13.10 (σ =3.62%)</td></tr>
<tr><td>Setpoint Uncertainty</td><td colspan="4">5.96% (1.645σ)</td></tr>
</table>

Table 2. DNBR sensitivities and uncertainties of the Yonggwang 3&4 nuclear power plant [2].

Parameters	Nominal value (m_i)	Standard deviation (d_i)	Sensitivity (S_i)	d_i/m_i
Primary coolant flow rate	1.0	0.025	1.3937	0.0250
Core power	1.0	0.01	-1.8789	0.0100
Core inlet temperature [^oF]	564.5	1.5	-8.1070	0.0027
Primary system pressure [*psia*]	2250	30	2.2852	0.0134
Nuclear enthalpy rise hot channel factor	1.55	0.0243	-1.2455	0.0157
Engineering enthalpy rise hot channel factor	1.0	0.015	-0.4492	0.0150
Engineering heat flux hot channel factor	1.0	0.015	-1.0021	0.0150
T/H code		0.025	1.0	0.0250

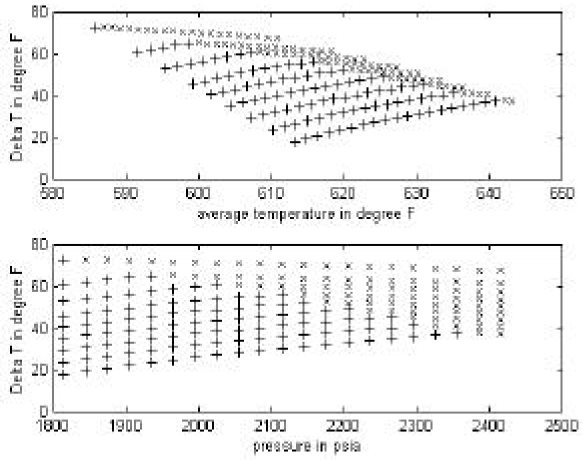

Fig. 3. Actual DNB data.

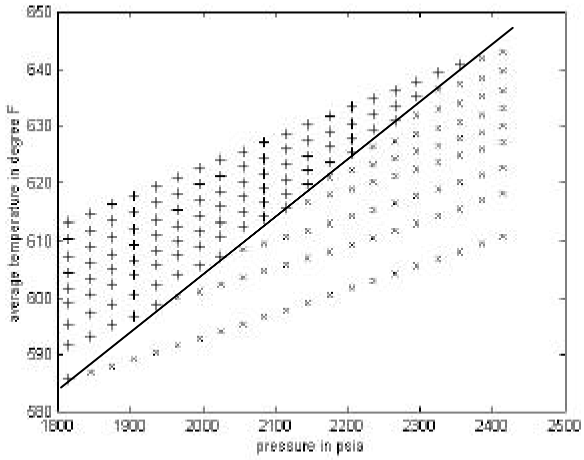

Fig. 4. Separation of the data region for two neurofuzzy inference systems.

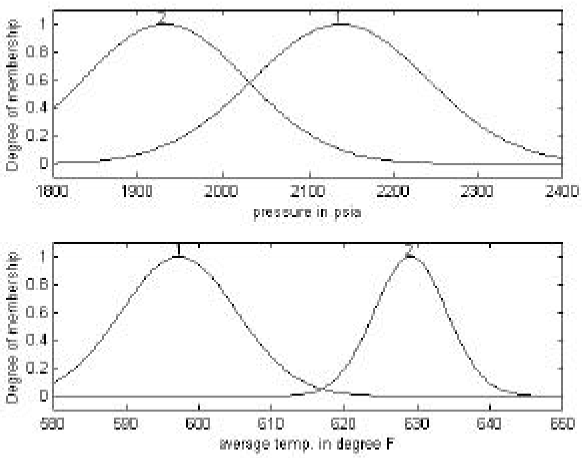

Fig. 5. Finally tuned membership functions for the first inference system.

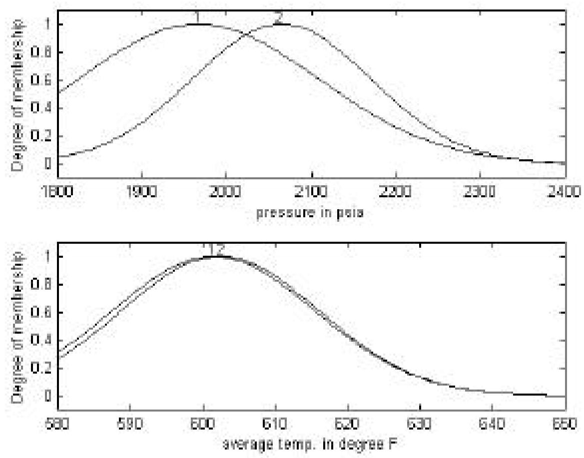

Fig. 6. Finally tuned membership functions for the second inference system.

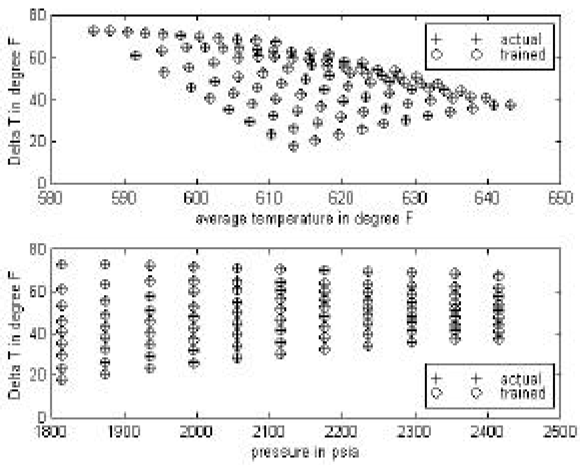

Fig. 7. Comparison of actual and trained ΔT .

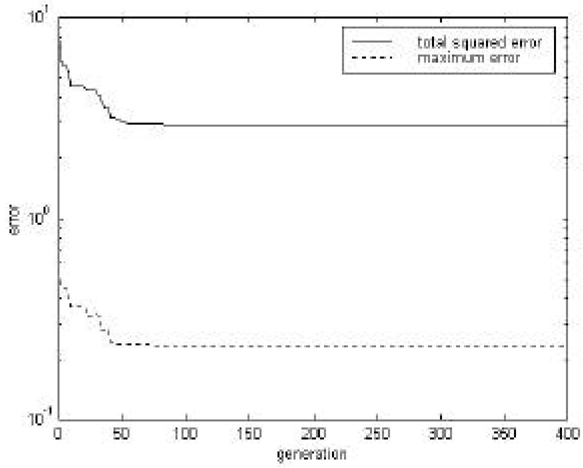

Fig. 8. Total squared error and maximum error.

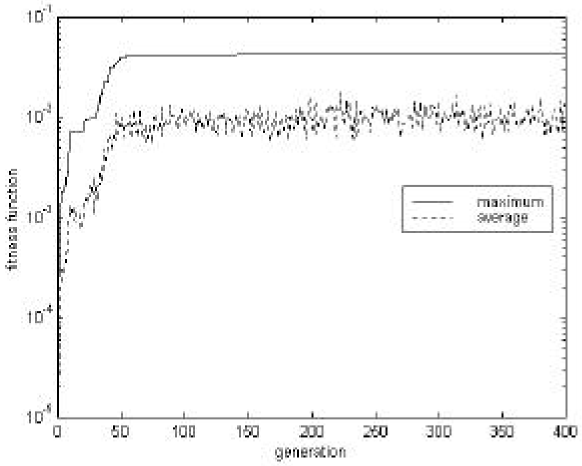

Fig. 9. Total squared error and maximum error.

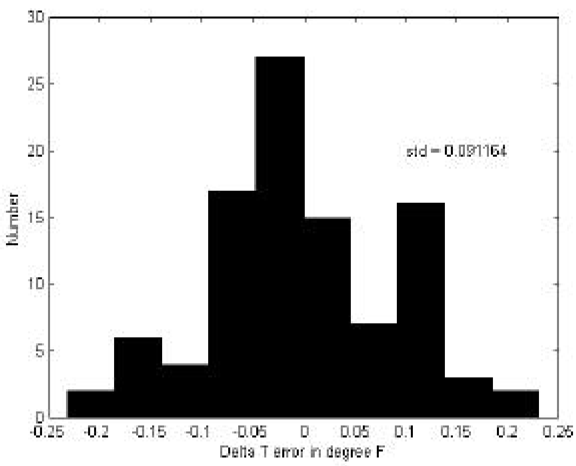

Fig. 10. Distribution of errors between the estimated and actual values.

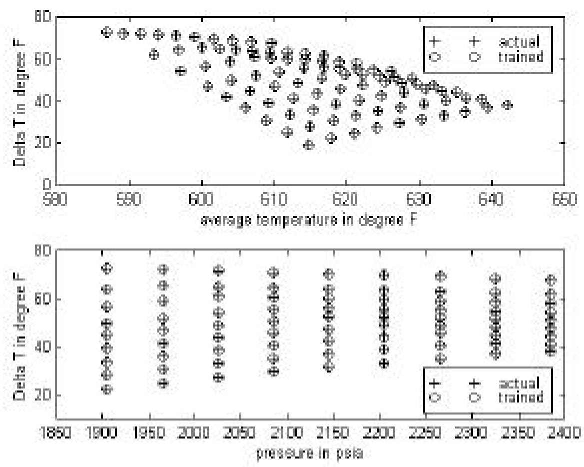

Fig. 11. Comparison of actual and trained ΔT using nontrained data.