Nuclear Engineering and Technology 53 (2021) 4014–4021

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Prediction of golden time for recovering SISs using deep fuzzy neural networks with rule-dropout

Hye Seon Jo [a], Young Do Koo [b], Ji Hun Park [a], Sang Won Oh [a], Chang-Hwoi Kim [b], Man Gyun Na [a, *]

[a] Department of Nuclear Engineering, Chosun University, 309 Pilmun-daero, Dong-gu, Gwangju, 61452, Republic of Korea
[b] Korea Atomic Energy Research Institute, 989-111 Daedeok-daero, Yuseong-gu, Daejeon, 34039, Republic of Korea

ARTICLE INFO

Article history:
Received 15 April 2021
Received in revised form
22 May 2021
Accepted 6 June 2021
Available online 11 June 2021

Keywords:
Golden time
Safety injection system (SIS)
Reactor core uncovery
Reactor vessel (RV) failure
Deep fuzzy neural network (DFNN)
Rule-dropout

ABSTRACT

If safety injection systems (SISs) do not work in the event of a loss-of-coolant accident (LOCA), the accident can progress to a severe accident in which the reactor core is exposed and the reactor vessel fails. Therefore, it is considered that a technology that provides recoverable maximum time for SIS actuation is necessary to prevent this progression. In this study, the corresponding time was defined as the golden time. To achieve the objective of accurately predicting the golden time, the prediction was performed using the deep fuzzy neural network (DFNN) with rule-dropout. The DFNN with rule-dropout has an architecture in which many of the fuzzy neural networks (FNNs) are connected and is a method in which the fuzzy rule numbers, which are directly related to the number of nodes in the FNN that affect inference performance, are properly adjusted by a genetic algorithm. The golden time prediction performance of the DFNN model with rule-dropout was better than that of the support vector regression model. By using the prediction result through the proposed DFNN with rule-dropout, it is expected to prevent the aggravation of the accidents by providing the maximum remaining time for SIS recovery, which failed in the LOCA situation.

1. Introduction

When a loss-of-coolant accident (LOCA), which is a design basis accident, occurs in nuclear power plants (NPPs), safety injection systems (SISs) operate to cool the reactor core and secure a shutdown margin through the injection of high-concentration boric acid water. However, if SISs do not operate in time during an accident, the core cooling capability is lost, which can lead to a severe accident in which reactor core uncovery and reactor vessel (RV) damage occur. Even if SISs are not normally activated at the beginning of an accident, reactor core uncovery and RV damage can be prevented by recovering the SISs in time during an accident. Therefore, to prevent the progression to a severe accident caused by a LOCA, a technique to predict the time for SIS recovery is considered necessary if SISs do not normally operate in an accident situation. In this study, the time mentioned above is defined as the golden time, that is, the maximum available time for SIS recovery to prevent reactor core uncovery and RV failure when SISs fail to operate at the beginning after LOCA occurrence. If the SIS is restored and normally operating within the golden time, the core uncovery and RV failure can be prevented.

The golden time was predicted using support vector regression (SVR) in a previous study [1]. In this study, a deep fuzzy neural network (DFNN) with rule-dropout is used to predict the golden time. DFNN is a method in which syllogistic fuzzy reasoning through multi-connected modules of fuzzy neural networks (FNNs) is simplified, and its performance is generally affected by the number of FNN modules and the nodes in the FNN. DFNN with rule-dropout used in this paper is a method in which the number of fuzzy rules for each FNN module is optimized to improve the performance of the existing DFNN [2–4]. The simulated data applied to develop the DFNN model with rule-dropout were acquired using the modular accident analysis program (MAAP) [5]. The MAAP code is a software tool that is used for accident sequence analysis in a referenced plant where an assumed accident scenario is applied, and presents an accident behavior as numerical values. In this study, the scenarios in which SISs do not operate at the initial time after LOCA occurrence in the optimized

* Corresponding author.
E-mail address: magyna@chosun.ac.kr (M.G. Na).

https://doi.org/10.1016/j.net.2021.06.017

pressurized reactor 1000 (OPR-1000) were postulated, and the simulated data for the corresponding circumstance were obtained and processed.

Many studies, such as break size estimation [6,7] RV water level prediction [8], and leak flow prediction [9] have been conducted to provide accident information to the operator and reduce human error using artificial intelligence (AI) methods in LOCA situations. Likewise, this study for golden time prediction was carried out for the same purpose. Therefore, the objective of this study is to develop a model that can help prevent a design-based accident situation from progressing to a severe accident by accurately predicting the SIS recovery time. In this paper, Section 2 describes the characteristics, training, and optimization of the DFNN with rule-dropout. Section 3 describes the postulated accident scenarios for data acquisition, data preprocessing, and model development. In Section 4, the prediction result of the DFNN with rule-dropout is presented and compared to the prediction performance of SVR of the previous study. Finally, Section 5 presents a summary and conclusions, as well as an assessment of the effectiveness of the DFNN with rule-dropout.

2. Deep fuzzy neural networks with rule-dropout

2.1. Rule-dropout

The main goal of this study is to develop a model with accurate prediction performance that can contribute to accident mitigation in the event of LOCAs in NPPs. To achieve this goal, the optimal algorithm should be selected from the data to be applied. In general, the efficacy of machine learning algorithms depends on its inherent characteristics; thus, a specific algorithm cannot be the best for all kinds of data. To select the optimal algorithm from the data to be applied, the data should be applied directly to all algorithms. However, because it takes a lot of time to apply the data to all algorithms, the applicable one is selected by preferentially applying several algorithms that are suitable for the applied data based on the characteristics of each algorithm.

Hence, as an AI technology, a method based on the existing DFNN was used. Although high-level prediction performance, which can be considered suitable for a domain, was derived from the existing DFNN model in previous studies [2–3], appropriate results were not always shown for all domains [4]. One of the reasons for this is that an optimal network structure is not deterministic according to the target domain or applied data. Therefore, we applied a dropout concept to DFNN so that the detailed structure of the existing DFNN can be properly adjusted during training to configure an optimal model in a domain to be applied. The DFNN with rule-dropout was used in the previous study to predict the internal states of the NPP containment [10]. This method showed better performance than the existing DFNN when predicting hydrogen concentration and containment pressure in LOCA situations.

The rule-dropout applied to DFNN in this study, which is similar to the dropout implemented in deep learning, can be explained as a method for preventing overfitting by controlling the number of nodes in the network that affects fuzzy inference performance. In general, dropout is a technique that randomly skips neurons with a specific probability during neural network training to prevent overfitting [11]. The skipped neurons are then temporarily deactivated in the neural network during training. Then, when the test is performed, the skipped neurons are revived, and the weight of each neuron is multiplied by the retained probability of each neuron to derive the final output value. Namely, this method prevents neurons from excessive co-adaptation during training [11].

Rule-dropout is a method for optimizing the number of fuzzy rules in an FNN module, which is the basic structure of DFNN. Specifically, the number of fuzzy rules to be deactivated is selected within the maximum number of fuzzy rules in an FNN module during training. The maximum number of fuzzy rules is the same for each module. In rule-dropout, the fuzzy rules to be deactivated are determined through a genetic algorithm (GA) [12]. The GA selects specific fuzzy rules considered inappropriate. When an FNN module is added, the nodes for the fuzzy rules to be activated are configured, and the nodes to be deactivated are removed. Unlike the dropout of deep learning, the nodes that are deactivated by rule-dropout are not reactivated and are permanently removed even in the test process. The optimal fuzzy rule number adjusted by the rule-dropout is the same as or different from each FNN module. Although the higher the number of fuzzy rules, the more in-depth fuzzy inference is possible; however, a model can be vulnerable to overfitting concurrently. Therefore, the rule-dropout characteristics allow performance to be improved gradually as it passes through the configured modules.

2.2. DFNN with rule-dropout

The existing DFNN, which is a fundamental architecture of the proposed DFNN with rule-dropout, has a structure in which more than two FNN modules are connected in series based on syllogistic fuzzy reasoning. That is, an inference result from the preceding step with an immediate connection is transmitted into the present step as an extra input. Because of this characteristic in the existing DFNN, high-level inference performance can be produced based on a relatively simplified syllogistic fuzzy inference [13]. The final output of the existing DFNN is calculated by using both initial inputs, and one additional input gradually improved through each added module; that is, the output of the final module determined in the latter part becomes the final result. These characteristics of the existing DFNN are inherited by the DFNN model with rule-dropout.

However, as mentioned above, the DFNN with rule-dropout is that the network structure of each module is severally adjusted, which is different from the existing DFNN. Because the FNN modules in the existing DFNN have the same fuzzy rule number, the nodes for n rules are equally configured in every module. Furthermore, fuzzy *if-then* operations are performed for all fuzzy rules in each module. However, in DFNN with rule-dropout, the nodes of the specific rules determined to be inappropriate among all n fuzzy rules are dropped; thus, the internal structure of each FNN module may be different or the same. Accordingly, the fuzzy inference operation is performed only on the surviving fuzzy rules. Eqs. (1) and (2) represent fuzzy *if-then* operations for the fuzzy rules activated in the existing DFNN and DFNN with rule-dropout, respectively;

$$
\begin{aligned}
&\text{for an arbitray i} - \text{th rule in the first module}\\
&\begin{cases}
\text{If } x_1(k) \text{ is } A_{i1}^{(1)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{im}^{(1)}(x_m(k)),\\
\text{then } \widehat{y}^{(1)}(k) \text{ is } f_i(x_1(k), \cdots, x_m(k))
\end{cases}\\
&\text{for an arbitray i} - \text{th rule from the 2nd to } g - \text{th module}\\
&\begin{cases}
\text{If } x_1(k) \text{ is } A_{i1}^{(g)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{im}^{(g)}(x_m(k))\\
\text{AND } \widehat{y}^{(g-1)}(k) \text{ is } A_{i(m+1)}^{(g)}(x_{(m+1)}(k)),\\
\text{then } \widehat{y}^{(g)}(k) \text{ is } f_i(x_1(k), \cdots, x_m(k), \widehat{y}^{(g-1)}(k))
\end{cases}
\end{aligned}
\tag{1}
$$

where $x_1, x_2, \cdots, x_m$ are input variables, $A_{i1}, A_{i2}, \cdots, A_{im}$ are the membership functions of all the inputs for the i-th fuzzy rule ($i = 1, 2, \cdots, n$) in the FNN module, $\widehat{y}$ is the output of each FNN module, f_i is the consequent for the i-th fuzzy rule, and k indicates the time instant of the applied data. $\hat{y}^{(g-1)}$, which is the $(m+1)$-th input, is an additional input for the i-th fuzzy rule from the second to the final g-th module.

Fig. 1 shows the structure of the proposed DFNN with rule-dropout in this study. In Fig. 1, the dotted squares and circles represent the nodes and their connections to the fuzzy rule dropped by rule-dropout, while the solid lines denote the maintained ones. In the existing DFNN, all of the nodes are depicted as solid squares, circles, and connecting lines without dotted shapes. In Eq. (2), the $(n$-1)-th fuzzy rule in the second module and the n-th fuzzy rule in the g-th module are deactivated, as shown in Fig. 1:

$$
\begin{aligned}
&\text{in the 2nd module}\\
&\left\{\begin{array}{l}
\text{If } x_1(k) \text{ is } A_{11}^{(2)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{1m}^{(2)}(x_m(k))\\
\text{AND } \hat{y}^{(1)}(k) \text{ is } A_{1(m+1)}^{(2)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(2)}(k) \text{ is } f_1(x_1(k),\cdots,x_m(k),\hat{y}^{(1)}(k))\\
\vdots\\
\text{If } x_1(k) \text{ is } A_{(n-2)1}^{(2)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{(n-2)m}^{(2)}(x_m(k))\\
\text{AND } \hat{y}^{(1)}(k) \text{ is } A_{(n-2)(m+1)}^{(2)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(2)}(k) \text{ is } f_{(n-2)}(x_1(k),\cdots,x_m(k),\hat{y}^{(1)}(k))\\
\text{If } x_1(k) \text{ is } A_{n1}^{(2)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{nm}^{(2)}(x_m(k))\\
\text{AND } \hat{y}^{(1)}(k) \text{ is } A_{n(m+1)}^{(2)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(2)}(k) \text{ is } f_n(x_1(k),\cdots,x_m(k),\hat{y}^{(1)}(k))
\end{array}\right.\\
&\vdots\\
&\text{in the } g-\text{th module}\\
&\left\{\begin{array}{l}
\text{If } x_1(k) \text{ is } A_{11}^{(g)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{1m}^{(g)}(x_m(k))\\
\text{AND } \hat{y}^{(g-1)}(k) \text{ is } A_{1(m+1)}^{(g)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(g)}(k) \text{ is } f_1(x_1(k),\cdots,x_m(k),\hat{y}^{(g-1)}(k))\\
\text{If } x_1(k) \text{ is } A_{21}^{(g)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{2m}^{(g)}(x_m(k))\\
\text{AND } \hat{y}^{(g-1)}(k) \text{ is } A_{2(m+1)}^{(g)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(g)}(k) \text{ is } f_2(x_1(k),\cdots,x_m(k),\hat{y}^{(g-1)}(k))\\
\vdots\\
\text{If } x_1(k) \text{ is } A_{(n-1)1}^{(g)}(x_1(k)) \text{ AND } \cdots \text{ AND } x_m(k) \text{ is } A_{(n-1)m}^{(g)}(x_m(k))\\
\text{AND } \hat{y}^{(g-1)}(k) \text{ is } A_{(n-1)(m+1)}^{(g)}(x_{(m+1)}(k)),\\
\text{then } \hat{y}^{(g)}(k) \text{ is } f_{(n-1)}(x_1(k),\cdots,x_m(k),\hat{y}^{(g-1)}(k))
\end{array}\right.
\end{aligned}
\tag{2}
$$

The number of fuzzy rules in DFNN with rule-dropout is severally adjusted in each FNN module to obtain high-level prediction performance while avoiding overfitting. To elicit high prediction performance through fuzzy rule operation, a five-layer network with a Takagi-Sugeno-type fuzzy inference system [14] is implemented in the FNN modules of DFNN with rule-dropout (refer to Fig. 1). The first layer in the five-layer FNN consists of nodes in which a membership function for fuzzy inference is defined. The results for each fuzzy rule are generated in the fourth layer. Finally, the final output of a single FNN module is calculated in the fifth layer.

2.3. Optimization of DFNN with rule-dropout

In addition to the fuzzy rule number, the parameters of each FNN module were optimized during the training for high-level prediction accuracy of the DFNN with rule-dropout used in this study. After the training was completed, the optimal number of modules was determined among the total configured modules, and then the model comprised of the optimal number of FNN modules was tested. The optimization procedure of the DFNN with rule-dropout is shown in Fig. 2. Once an FNN module is added, the fuzzy rule number and the membership function parameters c_{ij} and s_{ij} are determined by the GA [12] for optimal fuzzy inference in each module. The membership function is expressed in the form of a Gaussian function as follows:

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

where c_{ij} and s_{ij} are the center position and sharpness of the membership function, respectively.

The GA optimizes parameters to be suitable for a problem to be solved through the evolutionary process of an organism. Optimization through GA in DFNN with rule-dropout includes the process of generating and evolving populations of chromosomes that include the candidates for the parameters (e.g., number of fuzzy rules, c_{ij}, and s_{ij}) through genetic operation for multiple generations, and the process of evaluating each candidate and selecting the fittest solution using a fitness function. During the training of the DFNN model with rule-dropout, the candidate chromosome with the highest value among all the candidate chromosomes with the scores calculated through the fitness function is finally selected as the optimal solution in a single FNN module. The fitness function that evaluates the suitability of the FNN parameters in this study is as follows:

$$
\begin{aligned}
&F_{tr} = e^{-(\alpha E_t + \beta E_{t,\max} + \alpha(E_v - E_t) + \beta(E_{v,\max} - E_{t,\max}))}\\
&\text{constraints}\left\{\begin{array}{l}\alpha(E_v - E_t) = 0, \text{ if } E_v < E_t\\ \beta(E_{v,\max} - E_{t,\max}) = 0, \text{ if } E_{v,\max} < E_{t,\max}\end{array}\right.
\end{aligned}
\tag{4}
$$

where

$$E_t = \sqrt{\frac{1}{N_t}\sum_{k=1}^{N_t}\left(\frac{y(k)-\hat{y}(k)}{y(k)}\times 100\right)^2},$$

$$E_{t,max} = \max_k\left|\frac{y(k)-\hat{y}(k)}{y(k)}\times 100\right|, k=1,2,\cdots,N_t,$$

$$E_v = \sqrt{\frac{1}{N_v}\sum_{k=1}^{N_v}\left(\frac{y(k)-\hat{y}(k)}{y(k)}\times 100\right)^2},$$

$$E_{v,max} = \max_k\left|\frac{y(k)-\hat{y}(k)}{y(k)}\times 100\right|, k=1,2,\cdots,N_v.$$

α and β are the weighting coefficients for errors in the training and validation data, respectively. Additionally, N_t and N_v are the number of training and validation data, respectively.

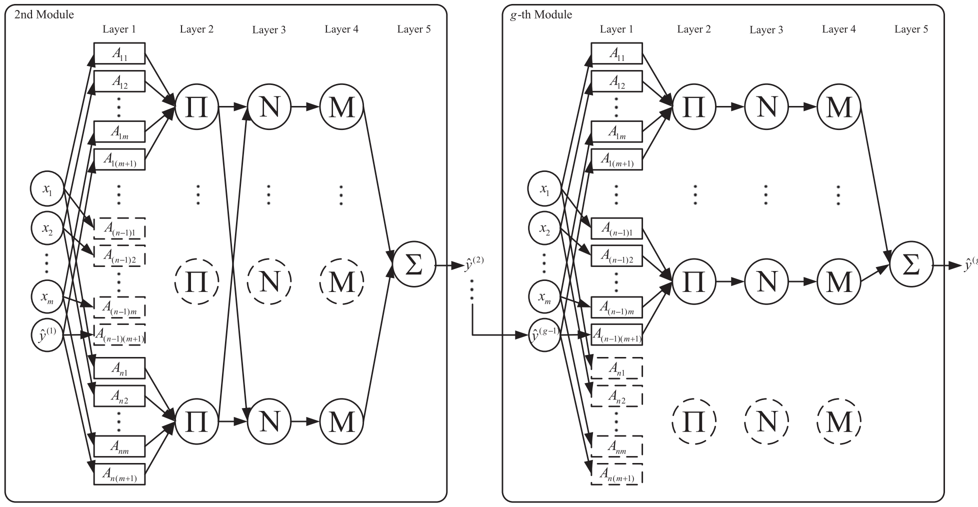

Fig. 1. Structure of the DFNN with rule-dropout.

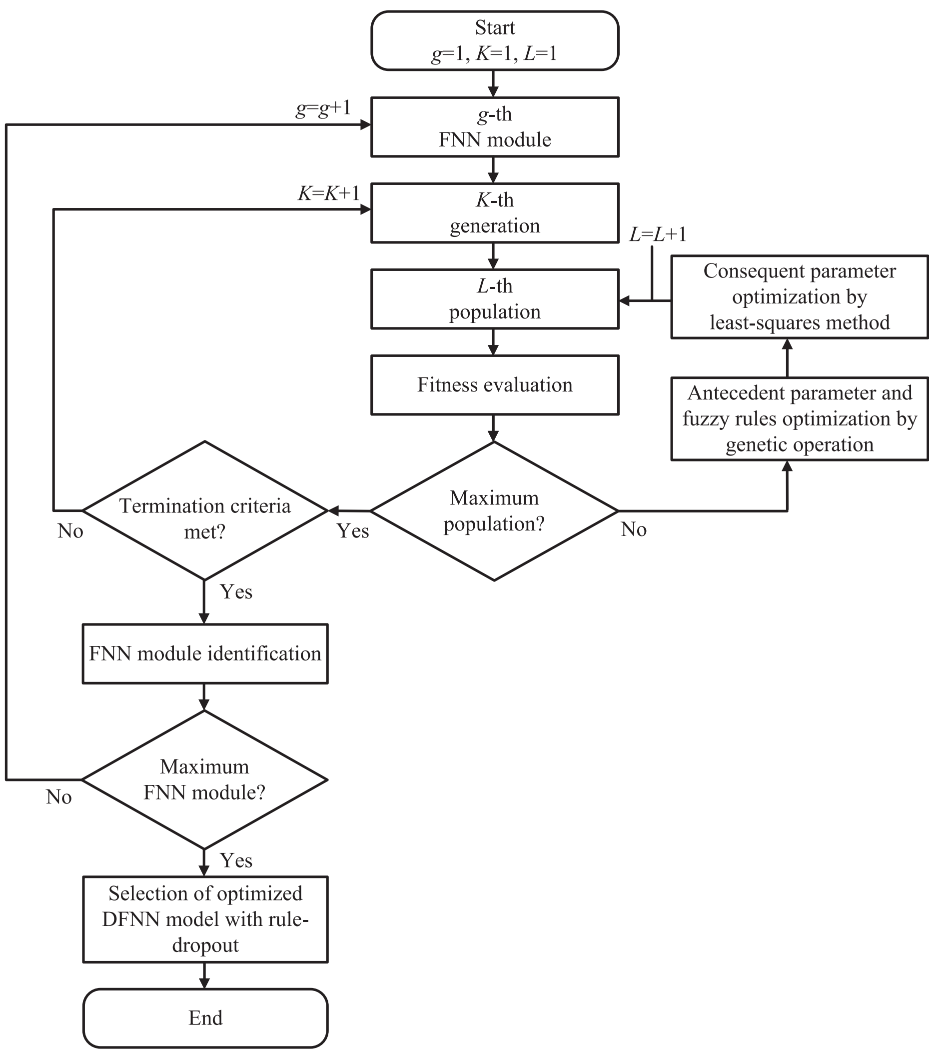

Fig. 2. Optimization procedure of the DFNN model with rule-dropout.

The output of each module is calculated by summing all the rule outputs f_i. Here, f_i is the consequent value for each fuzzy rule mentioned in Eq. (1) or (2), and is generally defined as a first-order polynomial of the inputs, as shown in Eq. (5).

$$f_i(x_1(k), x_2(k), \cdots, x_m(k)) = \sum_{j=1}^{m} q_{ij} x_j(k) + r_i \quad (5)$$

where q_{ij} and r_i are the weight values and bias, respectively. From the second to the g-th module, the $(m+1)$-th input is added to Eq. (5).

The least-squares method is a standard method that minimizes the sum of squared errors. Therefore, q_{ij} and r_i in Eq. (5) were determined to minimize the error between the predicted and real values, as defined in the objective function of Eq. (6).

$$J = \frac{1}{2} \sum_{l=1}^{N_t} (y_l(k) - \hat{y}_l(k))^2 \quad (6)$$

where $y_l(k)$ is a targeted output value and $\hat{y}_l(k)$ is a predicted output value.

All the FNN modules are optimized using the GA and least-squares methods. During training, the module identification process progresses using another fitness function of Eq. (7):

$$F_{all} = e^{-(\mu_1 E_a + \mu_2 E_{a,\max})} \quad (7)$$

where μ_1 and μ_2 are the weighting values for the root mean square (RMS) and maximum errors, respectively. E_a and $E_{a,\max}$ are the RMS and maximum errors for the development data, which consist of the training and validation data, respectively.

Module identification involves a process in which the fitness value from Eq. (7) of the present FNN module is compared with those of all the preceding FNN modules. When the number of FNN modules reaches the maximum module number, the training for the model construction is terminated. The maximum number of FNN modules is set to 15 in this study. After then, the optimal number of FNN modules is determined to construct the optimal DFNN model with rule-dropout. This optimal number is determined such that the fitness value of the present FNN module calculated using Eq. (7) is greater than that of all the preceding modules and is less than 95% of the highest fitness value. The final fitness value is less than 95% of the highest fitness value so that it reduces the number of FNN modules for the final optimal DFNN with rule-dropout. That is, it is to prevent an excessive stack of FNN modules.

3. Data preparation

3.1. Accident scenario

The scenarios selected to predict the golden time are the postulated hot- and cold-leg LOCA scenarios when SISs do not normally function. To simulate an accident using the MAAP [5] code, the break size was divided into 270 cases from 1/10,000 of the double-ended guillotine break (DEGB) to the DEGB. In addition, it is assumed that high-pressure safety injection (HPSI) and low-pressure safety injection (LPSI) do not operate at the beginning of the accident, but are actuated late, and the safety injection tank and containment spray system operate normally. If the pressure of the reactor coolant system does not fall below the maximum discharge pressure of the pump, the boric acid water is not injected even if the LPSI operates. In the case of a small break size, the pressure is not lower than the maximum discharge pressure of the pump; hence, the pressurizer power-operated relief valve (PORV) was set to be opened when LPSI was delayed.

3.2. Data acquisition

MAAP is the software used for simulating a postulated accident and analyzing the sequence of an accident; thus, the trend of a number of process variables according to the elapsed time of an accident can be presented in numerical values. The data for the golden time prediction obtained from MAAP are the simulated data for seven days after the accident occurrence, and consist of 16 process variables related to the reactor coolant system, containment, and steam generator. The delay operation times of the HPSI and LPSI were changed according to the interval of 10 s for each break size to find the maximum delayed operation time that does not cause core uncovery or RV failure. That is, the maximum delayed operation time (i.e., the golden time) is the maximum available time for SIS recovery so that core uncovery or RV failure does not occurs in LOCA situations. The simulated accident scenarios were classified into four cases according to HPSI and LPSI operations in the hot- and cold-leg LOCAs, as shown in Table 1, and a total of 1080 simulated data were acquired.

3.3. Data preprocessing

Data preprocessing for developing a golden time prediction model involves input variable selection, data classification, and data standardization. The input variables were selected according to their relationship to reactor core integrity and heat removal among the 16 variables acquired through the MAAP code simulations. Table 2 lists the selected input variables. The data for the training are reorganized into time-integrated data from the reactor trip to a specific time among the simulated data according to the sequence of accidents (refer to Eq. (8)). Specifically, the integral ranges in this study were 30, 60, or 150 s after the reactor trip.

$$x_j(k) = \int_{t_s}^{t_s+\Delta t} g_j(t)dt \quad (8)$$

where $g_j(t)$ is a simulated input signal, t_s is the reactor trip time, and Δt is the integration time span.

To effectively learn and verify the DFNN model with rule-dropout, the entire datasets were divided into training, validation, and test datasets. In each simulated accident case, the validation and test datasets corresponded to 10% of the entire datasets, and the training dataset accounted for 80% of all datasets. First, the test dataset was extracted from all the datasets according to the designated intervals. The training dataset was then extracted using the subtractive clustering (SC) method [15]. SC is a clustering algorithm that selects a cluster center based on the potential of a data point. This method calculates the potential for each data point and selects the data point with the highest potential as the center of the first clustering. The equation for obtaining the potential in the SC is as follows:

$$P_1(t) = \sum_{j=1}^{m} e^{-\frac{4\|x_t - x_j\|^2}{r_a^2}}, \; t = 1, 2, \cdots, m \quad (9)$$

where r_a is a cluster radius, x_t and x_j are data points for the remaining datasets, except for the test dataset.

In general, the potential value of each data point is revised based on the center and maximum potential value of the i-th clustering using Eq. (10). Likewise, after the i-th cluster is selected, the data point corresponding to the highest potential value among the modified potential values is selected as the $(i+1)$-th clustering

Table 1
Simulated accident scenario cases.

Case	Break position	PRZ PORV	HPSI operation	LPSI operation	No. of data
1	Hot-leg	Close	Delay injection & recirculation	N/A	270
2		Open	N/A	Delay injection & recirculation	270
3	Cold-leg	Close	Delay injection & recirculation	N/A	270
4		Open	N/A	Delay injection & recirculation	270

PRZ, Pressurizer; PORV, Power-operated relief valve; HPSI, High-pressure safety injection; LPSI, Low-pressure safety injection.

Table 2
Input variables for DFNN with rule-dropout.

No.	Input variable
1	Core exit temperature
2	Pressure in containment
3	Pressurizer pressure
4	Pressurizer water level
5	Collapsed water level
6	Unbroken side steam generator water temperature

center. When the modified potential values are determined, those for the data points near the i-th cluster center are significantly reduced, and the corresponding data points are less likely to be selected as the $(i+1)$-th clustering center.

$$P_{i+1}(t) = P_i - P_c e^{-\frac{4\|x_t - x_c\|^2}{r_b^2}}, \ t = 1, 2, \cdots, m \tag{10}$$

where r_b is the penalty radius for limiting the number of clusters to be created. x_c is the center of the i-th clustering, and P_c is the potential value of x_c.

The clustering center selection is repeated until the number of cluster centers is equal to the number of training data. Finally, the remaining data belong to the validation dataset after the test, and the training datasets are selected from among all the data.

Before distributing the datasets, the data were normalized to a normal distribution with a mean of 0 and a variance of 1. Standardization was used to prevent the model from being biased to specific data during training. The standardization equation is as follows:

$$z = \frac{x - \bar{x}}{\sigma} \tag{11}$$

where x is the input value, $\bar{x}$ is the mean value of the input values, and σ is the standard deviation.

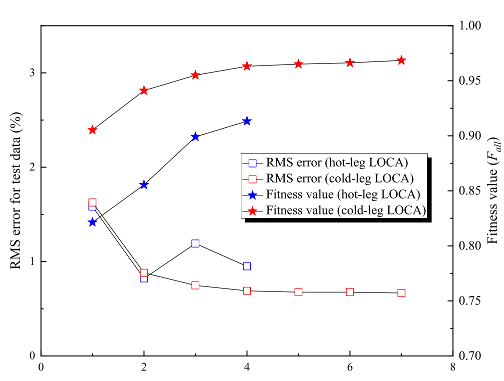

Fig. 3. Variation of RMS error and fitness values depending on the number of FNN modules (golden time prediction to prevent core uncovery in cases 2 and 4).

4. Golden time prediction result for SIS recovery

4.1. Golden time prediction result using DFNN model with rule-dropout

The golden time prediction for SIS recovery was performed using the optimal DFNN model with rule-dropout established through the model optimization process. The characteristic of the DFNN with rule-dropout is that as the FNN module and the fuzzy rule number increase, a deeper fuzzy inference is possible, and the performance is gradually improved. Fig. 3 shows this characteristic; specifically, it shows that as the FNN module is added, the RMS error decreases and the fitness value increases when predicting the golden time to prevent core uncovery in cases 2 and 4. That is, it can be seen that the performance progressively improves as the number of FNN modules is optimally set. In the developed DFNN model with rule-dropout, the fuzzy rule number optimized for each FNN module was between two and three when predicting the golden time for core uncovery prevention, and between three and four when predicting the golden time for RV failure prevention. This is because it was possible to accurately predict the golden time using the DFNN model with rule-dropout with only a small number of fuzzy rules in each applied domain, and overfitting occurred when a number of fuzzy rules were used.

Tables 3 and 4 show the results of predicting the golden time to prevent core uncovery and RV failure in all accident cases (i.e., cases 1–4 in Table 1). The DFNN with rule-dropout shows RMS errors within approximately 2.8% and maximum errors within approximately 8% for the test data when the HPSI operation is delayed in the LOCA (cases 1 and 3), respectively. In the case that LPSI operation is delayed (cases 2 and 4), RMS and maximum errors are within approximately 2.3% and 12% for the test data, respectively. Errors for prediction of the golden time to prevent RV failure are relatively higher than those for prediction of the golden time to prevent core uncovery. This is because the prediction error was comparatively higher in the smaller break size cases among the total break sizes. Specifically, in the data for the smaller LOCA break sizes, the change in the input values of the preprocessed data is small, but the predicted values fluctuate significantly; hence, it is difficult to predict accurately. Figs. 4–7 show the prediction results of the golden time to prevent core uncovery and RV failure for each accident case by using the DFNN model with rule-dropout.

4.2. Comparison of prediction results of DFNN with rule-dropout and SVR model

The golden time prediction performance of the proposed DFNN model with rule-dropout was compared with that of the SVR model used in a previous study [1]. SVR is a method based on the principle of structural risk minimization, which minimizes the upper limit for generalization errors rather than minimizing training errors; it maps input data to a high-dimensional feature space through a kernel function and determines the optimal regression function in the feature space [16]. The SVR model to be compared in the study

Table 3
Prediction performance of DFNN model with rule-dropout (HPSI delay).

Scenario	Prevention target	No. of FNN modules	Training data		Validation data		Test data	
			RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)
Hot-leg LOCA (case 1)	Core uncovery	8	1.705	5.265	2.251	5.057	2.784	6.688
	RV failure	12	3.659	26.475	0.566	1.023	1.878	7.405
Cold-leg LOCA (case 3)	Core uncovery	9	1.915	9.310	1.383	2.550	2.003	4.608
	RV failure	4	2.809	16.395	1.027	2.493	2.183	8.141

FNN, Fuzzy neural network; RMS, Root mean squares; LOCA, Loss-of-coolant accident; RV, Reactor vessel.

Table 4
Prediction performance of DFNN model with rule-dropout (LPSI delay).

Scenario	Prevention target	No. of FNN modules	Training data		Validation data		Test data	
			RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)
Hot-leg LOCA (case 2)	Core uncovery	4	0.827	2.093	0.564	1.039	0.950	1.959
	RV failure	5	0.912	5.314	0.521	1.035	2.373	11.756
Cold-leg LOCA (case 4)	Core uncovery	7	0.269	1.011	0.272	0.573	0.667	1.969
	RV failure	7	1.221	7.001	0.741	2.135	1.856	7.334

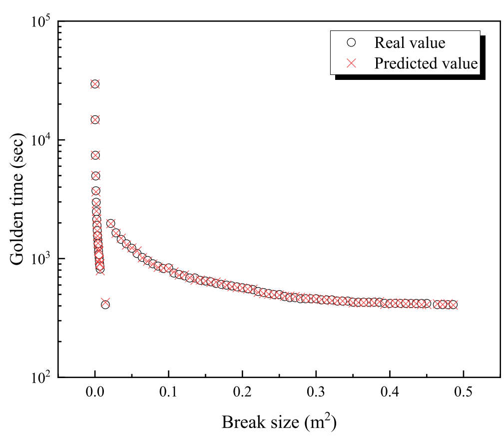

Fig. 4. Golden time prediction result to prevent core uncovery in case 1 using the DFNN model with rule-dropout.

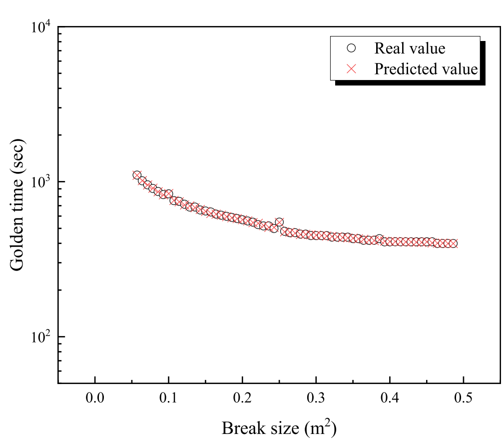

Fig. 5. Golden time prediction result to prevent core uncovery in case 2 using the DFNN model with rule-dropout.

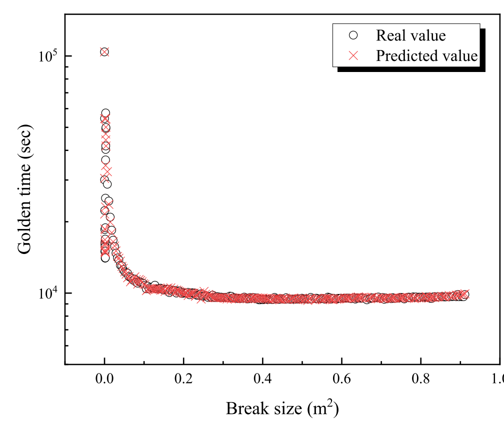

Fig. 6. Golden time prediction result to prevent RV failure in case 3 using the DFNN model with rule-dropout.

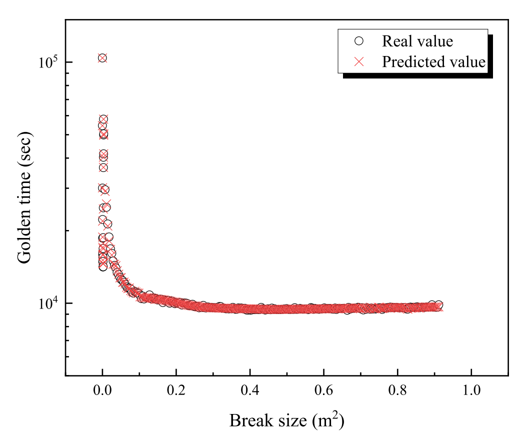

Fig. 7. Golden time prediction result to prevent RV failure in case 4 using the DFNN model with rule-dropout.

only has the same structure and features as the SVR model used in a previous study [1]. However, the applied data and input variables are not the same as those in the previous study [1], but are the same as the data applied to the DFNN model with rule-dropout in the present study.

Tables 5 and 6 show the results of predicting the golden time

Table 5
Prediction performance of SVR model (HPSI delay).

Scenario	Prevention target	Training data		Validation data		Test data	
		RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)	RMS error (%)	Max. error (%)
Hot-leg LOCA (case 1)	Core uncovery	0.621	2.854	1.957	4.648	2.947	4.936
	RV failure	7.336	82.357	0.576	1.250	1.957	7.137
Cold-leg LOCA (case 3)	Core uncovery	0.136	0.270	0.397	0.789	2.748	8.101
	RV failure	0.966	12.559	0.401	1.084	9.260	37.431

Table 6
Prediction performance of SVR model (LPSI delay).

Scenario	Prevention target	Training data		Validation data		Test data	
		RMS error (%)	Max.error (%)	RMS error (%)	Max.error (%)	RMS error (%)	Max.error (%)
Hot-leg LOCA (case 2)	Core uncovery	0.033	0.042	1.056	1.877	4.573	10.970
	RV failure	8.829	69.499	0.595	1.003	10.774	43.203
Cold-leg LOCA (case 4)	Core uncovery	0.235	1.355	0.377	0.937	3.000	8.566
	RV failure	1.229	15.105	0.923	3.028	8.818	43.952

using the SVR model. SVR has a feature that shows excellent generalization performance with only a small amount of data; however, in this study, it did not show high prediction performance in all cases. When predicting the golden time to prevent core uncovery, the proposed DFNN with rule-dropout shows better performance compared to the SVR. Also, when predicting the golden time to prevent RV failure, the proposed DFNN model with rule-dropout shows much lower RMS and maximum errors than the SVR model. The reason why the prediction performance of the DFNN model with rule-dropout is much better than that of the SVR model in the golden time prediction for RV failure prevention is considered to be a difference in the learning mechanism between the DFNN with rule-dropout and the SVR model. That is, unlike the learning mechanism of the SVR model, the DFNN model with rule-dropout can be accurately predicted because it derives the final output value through multi-stage learning. Multi-stage learning is a method that performs syllogistic fuzzy reasoning by using the result from the preceding step as an additional input for the present step with the addition of the FNN module. In addition, as the training time of the SVR model increases exponentially according to the number of training data, it takes slightly longer to train [16].

5. Conclusions

In this study, the DFNN model with rule-dropout was developed to predict the golden time for SIS recovery when the SISs failed in the initial phase in the LOCA situations. To develop the DFNN model with rule-dropout, the input values in the initial time after a reactor trip, and input variables related to reactor core integrity and heat removal in the simulated data acquired through MAAP were applied. GA and rule-dropout techniques were used to establish the optimal model by optimizing the FNN parameters and fuzzy rule numbers during training. Although the prediction performance of the developed DFNN model with rule-dropout was slightly lower than that of the SVR model in a few cases, it showed much better prediction performance, especially when predicting the golden time to prevent RV failure. The developed DFNN model with rule-dropout can accurately predict the golden time for accident mitigation in LOCA situations; therefore, it is considered that it can be utilized as a technology to alleviate accidents. Consequently, if the golden time for SIS recovery is provided when the SIS does not work in the LOCAs, the operator will be able to recognize the necessary information and act to recover the SIS in time.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (Grant Nos. NRF-2018M2A8A4025978 and 2018M2B2B1065651).

References

[1] K.H. Yoo, J.H. Back, M.G. Na, J.H. Kim, S. Hur, C.H. Kim, Prediction of golden time using SVR for recovering SIS under severe accidents, Ann. Nucl. Energy 94 (2016) 102–108.
[2] S.H. Yun, Y.D. Koo, M.G. Na, Collapse moment estimation for wall-thinned pipe bends and elbows using deep fuzzy neural networks, Nucl. Eng. Technol. 52 (2020) 2678–2685.
[3] Y.J. An, K.H. Yoo, M.G. Na, Y.S. Kim, Critical flow prediction using simplified cascade fuzzy neural networks, Ann. Nucl. Energy 136 (2020) 107047.
[4] H.S. Jo, Y.D. Koo, K.H. Yoo, M.G. Na, C.H. Kim, Prediction of NPP containment states using deep fuzzy neural networks during LOCAs, in: Proceedings of the Korean Nuclear Society Virtual Autumn Meeting, 17–18, 2020. December.
[5] EPRI, MAAP4 Applications Guidance, Final Report 1020236, Electric Power Research Institute, Palo Alto, CA, USA, 2010.
[6] M. Saghafi, M.B. Ghofrani, Real-time estimation of break sizes during LOCA in nuclear power plants using NARX neural network, Nucl. Eng. Technol. 51 (2019) 702–708.
[7] K.H. Yoo, Y.D. Koo, J.H. Back, M.G. Na, Identification of LOCA and estimation of its break size by multiconnected support vector machines, IEEE Trans. Nucl. Sci. 64 (2017) 2610–2617.
[8] Y.D. Koo, Y.J. An, C.H. Kim, M.G. Na, Nuclear reactor vessel water level prediction during severe accidents using deep neural networks, Nucl. Eng. Technol. 51 (2019) 723–730.
[9] J.H. Park, Y.J. An, K.H. Yoo, M.G. Na, Leak flow prediction during loss of coolant accidents using deep fuzzy neural networks, Nucl. Eng. Technol. 53 (2021) 2547–2555.
[10] Y.D. Koo, H.S. Jo, M.G. Na, K.H. Yoo, C.H. Kim, Prediction of the internal states of a nuclear power plant containment in LOCAs using rule-dropout deep fuzzy neural networks, Ann. Nucl. Energy 156 (2021) 108180.
[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (2014) 1929–1958.
[12] J. McCall, Genetic algorithms for modelling and optimisation, J. Comput. Appl. Math. 184 (2005) 205–222.
[13] J.C. Duan, F.L. Chung, Cascaded fuzzy neural network model based on syllogistic fuzzy reasoning, IEEE Trans. Fuzzy Syst. 9 (2001) 293–306.
[14] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Syst. Man Cybern. SMC- 15 (1985) 116–132.
[15] N. Dhanachandra, K. Manglem, Y.J. Chanu, Image segmentation using k-means clustering algorithm and subtractive clustering algorithm, Procedia Computer Science 54 (2015) 764–771.
[16] K.Y. Chen, C.H. Wang, Support vector regression with genetic algorithms in forecasting tourism demand, Tourism Manag. 28 (2007) 215–226.