- Technical paper -

DEVELOPMENT OF EVALUATION SYSTEM FOR REINFORCED CONCRETE BRIDGES

Bilal BAKHT[*1], Hiroshi MUTSUYOSHI[*2], Hayato TAKASE[*3]

ABSTRACT

The present study aims to develop an evaluation system for reinforced concrete bridges, constructed from multi-layer neural networks with fuzzy logic in order to carry out fuzzy inference and machine learning. The system evaluates the present performance of concrete bridge members in terms of serviceability, which is based on durability and load-carrying capability, using simple visual inspection and technical specifications. The neural network employed facilitates refinement of the knowledge base using back-propagation method, and prevents the inference mechanism of the system from becoming a black box. The results of the system when compared with the results of the bridge expert showed good agreement.

Keywords: serviceability, load-carrying capability, durability, fuzzy logic, expert system, neural network, knowledge-base.

1. INTRODUCTION

Since 1990, bridge maintenance costs have increased in several developed countries to the extent where it is now more expensive to maintain damaged bridges than to construct new ones. According to a report, in Japan, by 2010, approximately 35% of the bridges will be 50 years of age or older [1]. Therefore, the development of a comprehensive Bridge Management System (BMS) for exiting bridges has become essential. Such a system should enable, not only, the evaluation of bridge performance but also prediction of remaining life and suggestion of rehabilitation strategy taking into account the limited funds available both for construction and maintenance.

Kawamura et al [2] have developed an expert system that can evaluate the performance of concrete bridges using knowledge and experience acquired from the bridge experts. The objective of the present study is to develop an independent system with improved learning ability compared with the already developed system. The proposed system evaluates the present performance of bridge members in terms of factors such as serviceability, load-carrying capability, and durability. The input data for evaluating a concrete bridge are the technical specifications, environmental conditions, traffic volume, and other subjective information that can be obtained through simple visual inspection.

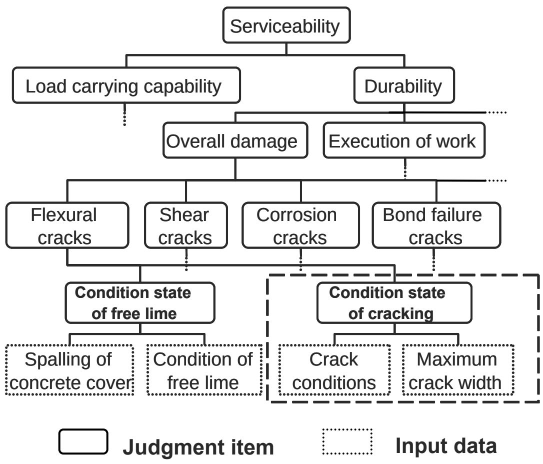

Fig. 1 Diagnostic process for main girders

The performance of a target bridge is evaluated according to a diagnostic process [2], which is modeled on the inference process used by domain experts for evaluating bridges (see Fig.1). This process is expressed by a hierarchical structure and has twelve upper rated judgment items and fifty lower rated judgment items. These judgment items are evaluated by about 90 input data items. The ultimate goal of this process is "serviceability." The relationship between judgment items and input data is expressed by "If-Then" rules with

*1 Master Student, Department of Civil Engineering, Saitama University, JCI Member
*2 Professor, Department of Civil Engineering, Saitama University, JCI Member
*3 Undergraduate Student, Department of Civil Engineering, Saitama University

fuzzy variables (fuzzy rules). In a computer, using Visual Basic, these rules have been transformed to a hierarchical neural network. The system refines the knowledge base by applying the Back-Propagation method [3]. The effectiveness of hierarchical neural network and back propagation was verified by comparison of the diagnostic results of bridge expert and proposed system.

2. PERFORMANCE EVALUATION

In Fig.1, the lower rated judgment items, such as "Condition state of flexural cracks around midspan of main girders" (Condition state of cracking) and "Condition state of free lime around midspan of main girders" (Condition of free lime), are first evaluated by use of input data. The final judgment item is "Serviceability", which is evaluated according to the results of "Load-carrying capability", and "Durability". Each of these judgment items is assigned a soundness score, on a scale of 0-100, which is output of the system. The output score is categorized into one of five groups: 0-12.5, 12.6-37.5, 37.6-62.5, 62.6-87.5 and 87.6-100 as shown in Table 1.

Table 1 Relation between soundness score & five categories

Category Soundness score	**Condition State Description**
Safe (87.6-100)	The bridge has no problem.
Fairly safe (62.6-87.5)	There is no serious damage.
Moderate (37.6-62.5)	There are some damages that require continuous inspection.
Slightly dangerous (12.6-37.5)	The bridge should be repaired and/or strengthened.
Dangerous (0-12.5)	The bridge should be removed from service and requires rebuilding.

3. KNOWLEDGE REPRESENTATION

In the knowledge base of the system, the diagnostic process is stored in the form of "If-Then" rules with fuzzy variables. Consequently, these rules enable the system to perform fuzzy inference. The knowledge representation of the system is as follows:

$$R^i : if\ x_1\ is\ A_1\ and \cdot\cdot and\ x_m\ is\ A_m\ then\ y\ is\ B_i \quad (1)$$

where, R^i : ith Fuzzy rule

$x_1, \cdots, x_m$: Input items (input data such as technical specifications and results of visual inspection)

y : Output item (judgment item)

$A_1, \cdots, A_m$: Fuzzy variables

B_i : Constant (soundness score on the scale of 0-100)

4. FUZZY INFERENCE PROCESS

This section describes in detail the fuzzy inference process performed in the system. The portion of Fig. 1 enclosed in a dotted box; namely, the inference process that evaluates "Condition state of cracking", is explained as an instance.

Table 2 Fuzzy rules for evaluating condition state of cracking

No	Antecedents		Consequent
	Crack conditions	Maximum crack width	Condition state of cracking Soundness score
1	Severe	Very large	Unsafe (0.0)
2	Severe	Large	Severe (25)
3	Severe	Small	Moderate (50)
4	Moderate	Very large	Severe (25)
5	Moderate	Large	Moderate (50)
6	Moderate	Small	Mild (75)
7	Not severe	Very large	Moderate (50)
8	Not severe	Large	Mild (75)
9	Not severe	Small	Safe (100)

Table 2 shows the fuzzy rules for evaluating the judgment item "Condition state of cracking". For example, Rule No. 9 expresses the following fuzzy rule; if ([Crack conditions] are {Not severe}) and ([Maximum crack width] is {Small}) then ([Condition state of cracking] is 100.0). Since the antecedents of the rules employ fuzzy variables, the initial form of membership functions for fuzzy rules have been prepared as fuzzy sets (see Figs. 2(a) & (b)). The membership functions for "Crack conditions" are singleton functions (see Fig. 2(a)) and for "Maximum crack width" are sigmoid functions (see Fig. 2(b)). Table 3 shows an

Table 3 Partial inspection sheet

G-1	**Cracking around midspans of main girders** (Flexural cracks)	■① Yes (go next) □② No (go to G-1.3)
G-1.1	Crack conditions	□① Severe ■② Moderate □③ Not severe
G-1.2	Maximum crack width	**0.3 mm**

excerpt of the inspection sheet used for the system. The solid squares indicate inspection results. The results of multiple choice questions are translated into numerical values and entered into the system. For "Crack conditions" the fuzzy variables "Severe," "Moderate," and "Not severe" correspond to input values 0.0, 0.5 and 1.0 respectively. Next, the inference process of "Condition state of cracking" diagnosis is described below, and is performed in 4 steps.

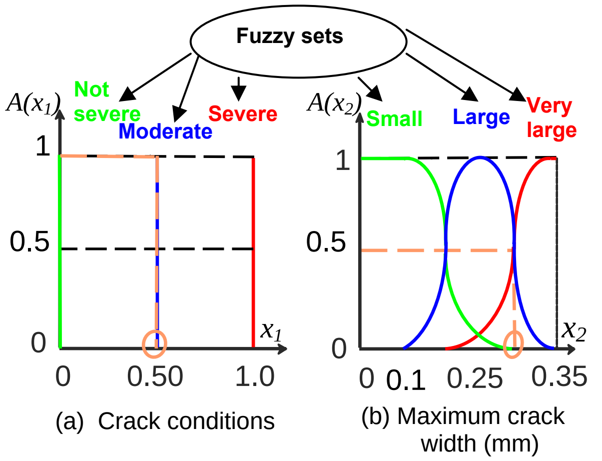

Fig. 2 Membership functions for input data

4.1 [Step 1] Data input

Input data are entered into the system. As shown in Fig. 1 the diagnosis of "Condition state of cracking" requires the input data [Crack conditions] and [Maximum crack width (mm)]. Therefore, the values of G1-1 and G1-2 in Table 3; that is "Moderate" and 0.3 mm, are used as the input data for the diagnosis.

No	Antecedents: Crack conditions	Antecedents: Maximum crack width		Fitness of rule (%)
1	Severe (0.0)	Very large (0.5)	(0.0) * (0.5) = 0.0	0.0
2	Severe (0.0)	Large (0.5)	(0.0) * (0.5) = 0.0	0.0
3	Severe (0.0)	Small (0.0)	(0.0) * (0.0) = 0.0	0.0
4	Moderate (1.0)	Very large (0.5)	(1.0) * (0.5) = 0.5	0.5 (50%)
5	Moderate (1.0)	Large (0.5)	(1.0) * (0.5) = 0.5	0.5 (50%)
6	Moderate (1.0)	Small (0.0)	(1.0) * (0.0) = 0.0	0.0
7	Not severe (0.0)	Very large (0.5)	(0.0) * (0.5) = 0.0	0.0
8	Not severe (0.0)	Large (0.5)	(0.0) * (0.5) = 0.0	0.0
9	Not severe (0.0)	Small (0.0)	(0.0) * (0.0) = 0.0	0.0
			Total sum = 1.0	

[Step 1 & Step 2] → [Step 3]

Fig. 3 Fuzzy inference process ①

4.2 [Step 2] Calculation of grades

Step2 calculates the grades of membership functions used in the antecedents of fuzzy rules. In this example, since the partial inspection sheet (see Table 3) shows multiple-choice value "Moderate" for the input data [Crack conditions], this multiple-choice value corresponds to the input value 0.5. This input value matches only one membership function that expresses the fuzzy set [Moderate] (see Fig. 2(a)). Therefore the grade of the membership function is 1.0 for the fuzzy set [Moderate]. Similarly, considering the inspection value of [Maximum crack width (mm)], which is 0.3, the value matches two membership functions, which express the fuzzy sets for [Large] and [Very large]. Therefore, these grades of membership functions are both 0.5 (see Fig. 2(b)). The left hand section table in Fig. 3 indicates the fitness of each fuzzy variable in antecedents to the inspection results; namely, [Crack conditions] =0.5 (Moderate) and [Maximum crack width (mm)] =0.3.

No	Fitness of rule		Consequent (Soundness score)	
1	0.0	*	0.0	= 0.0
2	0.0	*	25.0	= 0.0
3	0.0	*	50.0	= 0.0
4	0.5	*	25.0	= 12.5
5	0.5	*	50.0	= 25.0
6	0.0	*	75.0	= 0.0
7	0.0	*	50.0	= 0.0
8	0.0	*	75.0	= 0.0
9	0.0	*	100	= 0.0

(Input Data) Maximum crack width 0.3 mm & Crack condition 0.5 (moderate)

37.5

(Total sum)

Soundness score

→ [Step 4]

Fig. 4 Fuzzy inference process ②

4.3 [Step 3] Calculation of fitness of each rule

Step 3 calculates the fitness of each rule to input values. All fitness values of fuzzy variables in the same fuzzy rule are multiplied. Next their sum is calculated and finally each value is divided by the sum to calculate fitness of each rule (see Fig.3). For the input data entered, the values in the right-hand section in Fig. 3 are estimated. Rule No. 4 and Rule No. 5 both have a fitness of 50%.

4.4 [Step 4] Calculation of soundness score

In the final step, all fitness values of fuzzy rules are multiplied by the respective soundness score in the consequent and their sum is calculated. Hence for the input data entered, the system outputs the soundness score of 37.5 (see Fig. 4).

5. FUZZY INFERENCE BASED ON NEURAL NETWORK

In the expert system, the inference mechanism for evaluating a judgment item is constructed with a hierarchical neural network consisting of 4 layers for crack conditions and 5 layers for maximum crack width as shown in Fig. 5. The knowledge for diagnosing "Condition state of cracking"; that is to say, Table 2 and Figs. 2(a) & (b) (fuzzy rules and membership functions for fuzzy sets), are

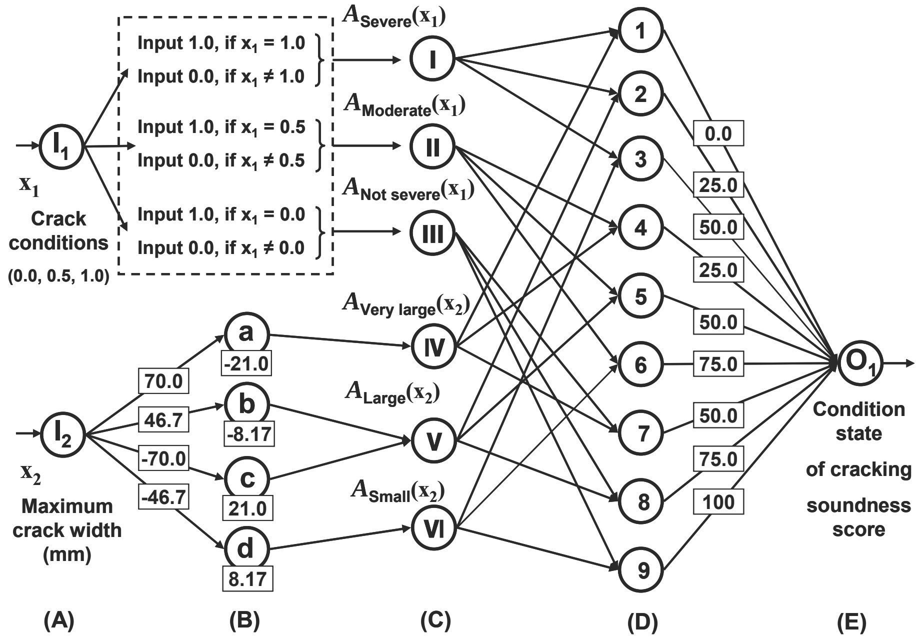

Fig. 5 Neural Network for evaluating "Condition state of cracking"

implemented in the computer by the neural network shown in Fig. 5. Therefore, the neural network can carry out the fuzzy inference mentioned in the previous section. In the present study, the layers of the network are referred to as layers (A), (B), (C), (D) and (E), respectively. These layers have neurons of three different types. The neurons in layers (A), (C) and (E) are linear neurons. The neurons in layer (B) are sigmoid neurons. The neurons in layer (D) are referred to as normalization neurons. The Arabic numerals in the layer (D) neurons correspond to the number (No.) in Table 2. Therefore, clearly the connections from layer (C) to layer (E) express a fuzzy rule. A boxed value represents the initial connection weight between neurons or the initial threshold for a neuron.

Next is described the manner in which the initial values of weight and threshold are set. The layers (A)-(B)-(C) in the network identify the fuzzy sets in antecedents of fuzzy rules for maximum crack width only. If the membership function of a fuzzy set is an increasing function or a decreasing function, the form is identified by a sigmoid function in layer (B). If the membership function is a convex function, the form is identified by the combination of two sigmoid functions in layer (B). For crack conditions since singleton functions have been used, which can have three input values only, thus behaving like crisp sets, layer (B) has not been used (see Fig. 5). The weights (ω) between layer (A) neurons and layer (B) neurons, and the thresholds (θ) of the (B) neurons are calculated according to the equations given in reference [2]. For increasing and decreasing functions the equations are the same as mentioned in the reference [2], however, for convex function the equation has been modified. The modified form of the equation is as follows.

① Approximation of convex function

$$\begin{cases} \omega_1 = h/A \\ \omega_2 = -h/C \\ \theta_1 = -h(B-A)/A \\ \theta_2 = +h(B+C)/C \\ x_1 = B, A(x_1) = 1.0 \\ x_2 = B-2A, A(x_2) = 0.0 \\ x_3 = B+2C, A(x_3) = 0.0 \\ x_1, x_2, x_3 \in X, x_2 < x_3 \end{cases} \tag{5}$$

Note that h is a real number, which satisfies $f(h) \approx 1.0$ where, $f(h)$: is sigmoid function. In the present study, $h = 3.5$. The weights between layer (B) neurons and layer (C) neurons are set to 1.0. In

Table 4 Partial questionnaire sheet

① What is the damage degree of flexural cracks?																				
Dangerous			**Slightly dangerous**					**Moderate**					**Fairly safe**					**Safe**		
0	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100

addition, initial weights between layer (C) neurons and layer (D) neurons are also set to 1.0. The initial weights between layer (D) neurons and layer (E) neurons are set according to Table 2. These weights express soundness scores described in consequent of fuzzy rules. Consequently, when input data are entered into the system, layers (A)-(B)-(C) perform the processing of [Step 1] and [Step 2] described earlier. Next, layers (C)-(D) perform the processing of [Step 3]. Finally, layers (D)-(E) perform the processing of [Step 4].

6. MODIFICATION OF FUZZY RULE BY MACHINE LEARNING

The elements modified by machine learning are the weights between layer (A) neurons and layer (B) neurons, the thresholds of layer (B) neurons, and the weights between layer (D) neurons and layer (E) neurons. The weights of layers (A)-(B) and the thresholds of layer (B) neurons are used in order to express membership functions in antecedents of fuzzy rules for maximum crack width only. Consequently, weight alteration after learning indicates the slope alteration of the corresponding membership function, and threshold alteration after learning indicates the axis movement of the membership function in the horizontal direction. In the learning of layers (D)-(E) weight, the proposition in consequent of fuzzy rules is changed. For instance, if the weight between a layer (D) neuron and a layer (E) neuron is changed from 10.0 to 20.0, the proposition described in consequent of fuzzy rule is changed from ([Condition state of cracking] is 10.0) to ([Condition state of cracking] is 20.0).

7. VERIFICATION OF EFFECTIVENESS OF MACHINE LEARNING

The proposed system is developed in Visual Basic programming language and runs on a personal computer. In this section, the system is applied to an existing bridge (one span), which is an RC T-girder-type bridge, in order to test validity of the learning capability. The target bridge is located in Saitama prefecture.

7.1 Questionnaire survey of domain expert and visual inspection of bridges

The purpose of the questionnaire survey of domain expert is to acquire teaching data necessary for learning (see Table 4); where as the purpose of visual inspection of bridge (see Figs. 6, & 7) is to collect inspection data to be entered into the system (see Table 3). The domain expert also needs the inspection results to fill out the questionnaire. The results of questionnaire survey and visual inspection were used as training data for carrying out machine learning.

Fig. 6 Side view of the bridge inspected

Fig. 7 Main girder deterioration

7.2 Practical application and verification of the evaluation system

A set of survey forms, prepared for the bridge in Fig. 6 consisted of three different handouts; inspection record sheets to be used to record visual inspection results, a model drawing of the bridge on which the respondent wrote down whatever came to his mind during inspection, and questionnaire sheets to obtain teaching data required for machine learning.

Table 5 summarizes the results of the main girder diagnosis by the system (DR) and domain expert (TD) in addition to the learned results (LR) and errors before and after learning. Twelve upper judgment items have been listed to show the effectiveness of machine learning, however, in all there are sixty two judgment items including upper and lower. The letters S, F-s, M, S-d, and D in the parentheses represent Safe, Fairly safe, Moderate, Slightly dangerous, and Dangerous. These labels classify the values given in the table into five

Table 5 Results of main girder diagnosis

Judgment items	DR	TD	Error	LR	TD	Error
Girder design	66.02 (F-s)	90 (S)	23.98	89.90 (S)	90 (S)	0.10
Girder execution	75.00 (F-s)	40 (M)	35.00	40.09 (M)	40 (M)	0.09
Service condition	87.73 (S)	90 (S)	2.27	90.08 (S)	90 (S)	0.08
Material deterioration	66.33 (F-s)	40 (M)	26.33	40.10 (M)	40 (M)	0.10
Flexural cracks	87.50 (S)	30 (S-d)	57.50	30.10 (S-d)	30 (S-d)	0.10
Shear cracks	100.0 (S)	30 (S-d)	70.00	30.09 (S-d)	30 (S-d)	0.09
Corrosion cracks	77.97 (F-s)	30 (S-d)	47.97	30.08 (S-d)	30 (S-d)	0.08
Bond cracks	100 (S)	100 (S)	0.00	100 (S)	100 (S)	0.00
Whole damage	85.99 (F-s)	55 (M)	30.99	55.10 (M)	55 (M)	0.10
Load-carrying capacity	49.71 (M)	90 (S)	40.29	89.90 (S)	90 (S)	0.10
Durability	82.91 (F-s)	50 (M)	32.91	50.08 (M)	50 (M)	0.08
Serviceability	71.40 (F-s)	80 (F-s)	8.60	79.90 (F-s)	80 (F-s)	0.1
Total error	Before learning		365.84	After learning		1.02

DR: Diagnosed Results, TD: Teaching Data, LR: Learned Results

categories, the criteria used by the respondent for this categorization having been mentioned earlier.

For the bridge inspected, the total error before learning is 365.84, however after learning the error reduces drastically to 1.02. For the individual judgment item, no error is more than 0.1. This proves the effectiveness of machine learning. However, since the reliability of the system depends on information on the distribution of bridge damage used for neural network learning, we must increase the number of sample bridge data sets used for learning and acquire data sets for various damage conditions.

8. CONCLUSIONS

In the present study, an evaluation system for reinforced concrete bridges has been developed independently, with improved learning ability compared with the system developed by Kawamura et al [2]. The diagnostic tree (see Fig. 1) is the same as used by Kawamura et al [2], however, machine learning algorithms have been written for all the judgment items (sixty two in number); unlike Kawamura et al [2], who connect the input data items directly with the upper judgment items (eleven in number for their system). The approach adopted in the present study increased the task in terms of writing the code and collecting teaching data for all the judgment items; however, the errors between the teaching data and system's output after learning, reduced drastically compared with the system developed by Kawamura et al [2].

The developed system not only performs fuzzy inference but also facilitates refinement of the knowledge base, based on data such as inspection results and questionnaire survey of domain expert. Moreover, the proposed neural network contributes to prevent the inference mechanism from becoming a black box. The system was applied to the main girders of existing bridge in order to verify the effectiveness of machine learning method. The knowledge base was refined from the results of questionnaire survey of domain expert. Good agreement between the target values and the values after learning confirms the effectiveness of the learning method in the system. In order to enhance the reliability of the expert system, the knowledge base must be refined through application to a greater number of bridges with various damage conditions.

ACKNOWLEDGMENTS

The authors would like to acknowledge Professor Ayaho MIYAMOTO and Research Associate Kei KAWAMURA for their support during the system's development.

REFERENCES

[1] Miyamoto, A., Kawamura, K., and Nakamura, H., "Development of a Bridge Management System for Existing Bridges," J. of Advances in Engineering Software, Elsevier, Vol. 32, 2001, pp. 821-833.

[2] Kawamura, K., and Miyamoto, A., "Condition state evaluation of Existing Reinforced Concrete Bridges Using Neuro-Fuzzy Hybrid System," J. of Computers and Structures, Elsevier, Vol. 81, 2003, pp. 1931-1940.

[3] Rumelhart, D.E., Hinton, G.E., Williams, R., J., "Learning Representations by Back Propagating Errors," Nature, Vol. 323(9), 1986, pp. 533-6.