Artificial intelligence-based novel design charts for doubly reinforced concrete beams

Won-Kee Hong, Van Tien Nguyen and Manh Cuong Nguyen

Department of Architectural Engineering, Kyung Hee University, Yongin Republic of Korea

1. Introduction

AI concepts have been used with successful outcomes in many studies for the area of structural analysis, including interesting studies as follows. Aberbres and Lantsoogt (2020) successfully implemented artificial neural networks (ANNs) to predict shear capacity of one-way slabs under concentrated loads. Lu et al. (2020) applied machine learning (ML) techniques such as tree predictive models and a novel feature selection to predict the punching shear capacity of steel fiber-reinforced concrete flat slabs. Moreover, Asteris et al. (2019) and Armaghani et al. (2019) also employed ANN efficiently in computing shear strength of RC beams. Sharifi, Lotfi, and Moghbeli (2019) applied ANN method to solve structural engineering problems. They predicted compressive capacity of fiber reinforcement polymer (FRP)-confined reinforced concrete (RC) columns based on ANNs. Charalampakis and Papanikolaou (2021) investigated the effect of a large number of training datasets on accuracies for training and design of RC columns, bridge piers by applying ML techniques. Flood and Kartam (1994a, 1994b) investigated the influence of several hidden layers and hidden nodes on the validation of a network and processing speed for deep learning networks. They concluded that the success of neural network implementation depends on the quality of the data used for training (The MathWorks 2020b). This study also emphasized the importance of training data for accurate design results. Lee et al. (2018) analyzed recent deep learning methods to apply innovative algorithms on structural analysis problems which is followed by...
Anderson et al. (1997) who performed structural analysis and design of steel connections. They predicted the design moment resistance and secant stiffness for minor axis steel connections by using an ANN that is based on network inputs, including depth of section, flange and web thickness for the column, flange breadth and depth of section for the beam, number of bolts, and plate thickness for the connection plate. Dahou et al. (2009) also proposed an ANN technology to model a bond between conventional ribbed steel bars and concrete. They explored the influence of a different number of inputs parameters on predicting an ultimate pull-out load between rebar and concrete. Recently, Rafiei and Adeli (2017) presented a neural model for the global health monitoring of large structures, such as high-rise building frames, by integrating two signal processing techniques based on an unsupervised ML technique. ANNs continue to gain interest in civil engineering applications. They have been used to solve numerous structural issues not only for analysis and design (Das and Choudhury 2019; Yaseen et al. 2018; Chen et al. 2018; Lee and Lee 2014; Azim et al. 2020; Allahyari et al. 2018; Huang et al. 2021; Bka et al. 2021; Asteris and Mokos 2020; Armaghan and Asteris 2020) but also for structural damage assessment (Neves et al. 2017; Oh et al. 2017; Zarbaf et al. 2018; Morfidis and Kostinakis 2018). However, Gupta and Sharma (2011) stated weak performance and numerical instability in training big structural model prevented from using neural network.

Many of AI-based studies were involved with structural analysis with particular research targets as shown in the references shown in this study. The present study intends to present the AI which can be implemented with design issues. The goal of this study is to overcome the predicament of some design issues with multiple design parameters, recognizing ANNs can learn the trends of big structural design datasets between inputs and outputs for engineering applications. In these efforts, input and output parameters considered in the ANN developed in this study can be exchanged freely for practical applications.

The network inputs that must be considered for the design of structural frames, in general, and doubly reinforced concrete beams, in particular, are material properties [yield strength of rebar \( f_y \) (MPa) and concrete compressive strength \( f_c \) (MPa)], beam geometry [beam width \( b \) (mm), height \( h \) (mm), depth \( d \) (mm), span length \( L \) (m)]. Networks also estimate the manufacturing beam cost, optionally. Input reverse parameters are design moment strength excluding moment due to self-weight \((\phi M_o)\), factored moment considering moment due to dead load and live load \((M_o \text{ of } 1.2M_o+1.6M_i)\) and curvature ductility \(\mu_\phi\). Network outputs include rebar ratios for both tension \(\rho_{tr}\) and compression \(\rho_{cr}\), strains of rebars in tension \((\varepsilon_{t,0.003})\) and compression \((\varepsilon_{c,0.003})\) at concrete strain of 0.003, immediate \(\Delta_{mm}\) and long-term \(\Delta_{long}\) beam deflections, and cost index of beam \(C_i\). This study aims to develop design of doubly reinforced concrete beams based on ANN. As many as needed, network inputs and outputs can be used. However, more network inputs and outputs can be adopted depending on the analysis, design, and computation power.

2. Research significance

ANN has been applied widely to perform forward structural designs as a surrogate tool of conventional design methods. It is noted that forward designs can be performed by using conventional software and ANN. However, it is time-consuming and challenging for conventional design methods to conduct reverse designs in which, for instance, design moment \(\phi M_o\) and curvature ductility \(\mu_\phi\) are constrained as input parameters. Conventional methods should be performed repeatedly until reverse designs of a ductile beam section are achieved with reverse input

![Figure 1. Analogy between a biological neuron model and an artificial neural network (ANN).](image-url)
parameters being given on input side. In this study, ANN with a capability to learn the trend of big datasets was implemented to overcome this issue by reversely preassigning governing design factors (ϕM₀ and μ₀) in input layer to predict beam dimensions and rebar ratios. Design charts based on ANN were also developed to assist engineers to design a ductile RC beam in the preliminary stage.

3. Deep neural networks for structural engineering

Electrical input activates biological neurons to send out pulses through their axons (Figure 1). Learning and memory capabilities similar to human brains are exhibited in an ANN. Voltage spikes along their axons are biologically transmitted to a part of the body associated with the specific neuron by collecting electro-mechanical signals between their dendrites.

ANNs are trained to map network inputs to outputs, which are then recalled for a vector of solutions (design results) to design without the knowledge of structural mechanics. A large number of interconnected nodes and deep hidden layers similar to biological neurons were developed to integrate all incoming signals, mapping the training results onto network outputs. Weighted interconnections and bias linking nonlinear numerical computations are then activated by a function as shown in Eq. (1), such as the rectified linear unit (ReLU) function. Network inputs (x₁ to xₙ) are transmitted to neurons of successive layers that are fully connected through weights at each neuron and bias in each hidden layer.

\[ y = \sigma^{n}(W^{n}a^{n-1}(W^{n-1} \ldots a^{1}(W^{1}x + b^{1}) \ldots + b^{n-1}) + b^{n}) \]  

(1)

In which, y is output parameter; x is input vector; N is a number of layers including hidden layers and output layer; Wⁿ is matrix of weight values between layer n-1 and n; bⁿ is bias values of layer n; \( \sigma^{n} \) is the activation function of layer n.

In this study, we developed deep neural networks that can learn trends from big structural design datasets generated from structural mechanics of ductile concrete beams to design doubly reinforced concrete beams for engineering applications rather than basing it on structural mechanics or knowledge. Close correlations between untrained test data and design values based on structural mechanics were validated.

4. Generation of large structural datasets and network training

The strain compatibility-based algorithm (Autobeam) developed for a ductile design of doubly reinforced concrete beams by Nguyen and Hong (2019) was used to generate a large structural datasets to train networks. Table 1 summarizes the ranges of the big structural datasets generated based on the Autobeam algorithm as shown in Figure 2(a). The random datasets for a rebar yield strength \( f_{y} \) and concrete compressive strength \( f_{c} \).

### Table 1. Range of parameters, cost index, and design scenarios.

(a) Range of parameters.

| Number of datasets | 100,000 |
|-------------------|---------|
| **Maximum**       | 12000   |
|                   | 1500    |
|                   | 1446    |
|                   | 1195    |
|                   | 50      |
|                   | 600     |
|                   | 0.05    |
|                   | 0.025   |
|                   | 23229   |
|                   | 11588   |
| **Average**       | 10000   |
|                   | 951     |
|                   | 877     |
|                   | 520     |
|                   | 40      |
|                   | 550     |
|                   | 0.0152  |
|                   | 0.0044  |
|                   | 1751    |
|                   | 538     |
| **Minimum**       | 8000    |
|                   | 400     |
|                   | 292     |
|                   | 120     |
|                   | 30      |
|                   | 500     |
|                   | 0.0023  |
|                   | 7.6E-06 |
|                   | 4       |
|                   | 0       |
| **Variance (σ²)** | 1370224 |
|                   | 101905  |
|                   | 96832   |
|                   | 51892   |
|                   | 37      |
|                   | 851     |
|                   | 9.4E-05 |
|                   | 1.7E-05 |
|                   | 5212456 |
|                   | 672151  |
| **Standard deviation** | 1171 |
|                   | 319     |
|                   | 311     |
|                   | 228     |
|                   | 6       |
|                   | 29      |
|                   | 9.7E-03 |
|                   | 4.2E-03 |
|                   | 2283    |
|                   | 820     |

(b) Forward and Reverse scenarios

| Design scenarios | Input | Forward | Reverse | Output |
|------------------|-------|---------|---------|--------|
| L                | h     | φM₀     | L₀      | φM₀    |
| d                | b     | μ₀      | d₀      | μ₀     |
| fₜ              | fₚ    | Mₚ      | fₚ₀     | Mₚ₀    |
| ρₜ              | ρₚ    | L₀      | ρₚ₀     | L₀₀    |
| M₀              | Mₚ₀   | d₀₀     | Mₚ₀₀    | d₀₀₀   |

*Red text: Reverse input parameters
**Blue text: Reverse output parameters

(c) Cost index of concrete and rebar

| Cost index | Concrete (KRW/m³) | Rebar (KRW/kgf) |
|------------|------------------|-----------------|
| fₚ = 30 MPa | 85,000           | 1055            |
| fₚ = 40 MPa | 94,000           | 1085            |
| fₚ = 50 MPa | 104,000          |                 |
were generated in the range of 500–600 MPa and 30–50 MPa, respectively, considering a wide range of beam designs for beam lengths spanning 8–12 m. Beam heights were considered between 400 and 1500 mm, whereas beam widths were (0.3–0.8) times beam height. Tensile and compressive rebar ratio were created in the range of 0.0023–0.05 and 7.6E-6 – 0.025, respectively. In this study, multiple rebar layers were also taken into consideration. Moment due to dead load and live load were also considered. Table 1 indicates maximum, average, minimum, variance, and standard deviation of random parameters \( L, h, d, f_\sigma, f_\tau, \rho_r, \rho_c, D, M_0, M_1 \). Cost index investigated in this study considered material and manufacturing costs of concrete and rebar as presented in Table 1. Figure 2(b) shows network algorithm by which ANN was trained on big structural data for optimized designs of ductile beams. Big structural datasets generated based on this algorithm were used to train ANN.

Networks are, then, established for concrete beam design as shown in Figure 1 where layers are then formulated as based on nine inputs \( L, b, f_\sigma, f_\tau, M_0, M_1, M_0, M_1, \phi, M_0, \mu_\phi \) including three reverse inputs \( (M_0, M_1, \phi, M_0, \mu_\phi) \) and nine outputs \( (h, d, \rho_r, \rho_c, \varepsilon_{c,0.003}, \varepsilon_{\psi,0.003}, \Delta_{imme}, \Delta_{long}, \Delta_{lub}) \) including four reverse outputs \( (h, d, \rho_r, \rho_c) \) as reverse design shown in Table 1.

The reverse design was performed to design a ductile reinforced concrete beam in which design moment \( (\phi M_0) \) was optimized to be equivalent to factored moment \( (M_0) \). Curvature ductility \( (\mu_\phi) \) was also chosen for a ductile section; tensile rebar strain \( \varepsilon_{\psi,0.003} \) is larger than or equivalent to 0.005 in according to definition of tension-controlled section of ACI 318-14. Training with combined three types of layers and three types of neurons determined best number of layers and neurons based on training accuracies.

5. Design of doubly reinforced concrete beams based on artificial neural network

5.1. Design scenarios

In this study, the authors considered only flexural design of doubly reinforced concrete beam in terms of fixed-fixed boundary conditions by applying reverse design techniques. Shear design and inelastic deflections of fixed-fixed or simply supported reinforced concrete beam will be considered in further research.
Reverse techniques can be implemented in many areas of design where sequences of calculating inputs and outputs are to be switched. Reversing design inputs and outputs is challenging for conventional designs, which can only validate when designs are completed. Table 1 presents design scenarios with a description of parameters implemented in forward and reverse designs. There are 10 inputs \((L, h, d, b, f_p, f_y, \rho_{rt}, \rho_{rc}, M_D, \text{ and } M_L)\) and eight outputs \((\phi M_n, M_u, \epsilon_{rt\cdot0.003}, \epsilon_{rc\cdot0.003}, \Delta_{imme}, \Delta_{long}, \mu_\phi, \text{ and } Cl_b)\) for forward design scenario. The forward design can be performed by conventional design methods. However, by conventional design methods, it is challenging to perform reverse designs in which input parameters including design strength \((\phi M_n)\) equal to factored moment \((M_n)\) and curvature ductility \((\mu_\phi)\) are pre-assigned in input while eight output parameters including four parameters \((h, d, \rho_{rt}, \text{ and } \rho_{rc})\) are calculated in output. These types of reverse designs would prescribe curvature (strain) ductilities in design to seek well-balanced...
frames sections, targeting factored moment. This study introduces noble methods based on artificial intelligent networks (ANN), offering solutions for reverse designs shown in Table 1.

5.2. Design of doubly reinforced ductile concrete beam

5.2.1. Forward design; formulation of artificial neural networks

Table 1 describes parameters used in design scenarios shown in Table 1. This section is devoted to the applications and verifications of artificial intelligence-based designs of fixed doubly reinforced concrete beams (Figure 3). Table 1 presents design scenarios including both forward and reverse designs. Forward designs can be performed by conventional design method whereas finding solutions of reverse designs (refer to Table 1) in which network inputs and outputs are exchanged are challenging via conventional design method. Forward design is based on ten input parameters (including beam material properties ($f'_c$, $f'_y$), beam geometries ($b$, $h$, $d$, $L$), tensile rebar ratio ($p_{rn}$), and compressive rebar ratio ($p_{rc}$)) and eight output parameters (including factored moment ($M_w$), design strength ($\phi M_n$), compressive and tensile strains ($\varepsilon_{rc,0.003}$ and $\varepsilon_{cr,0.003}$) corresponding to concrete strain of 0.003, curvature ductility ($\mu_c$), and $C_{lb}$ (cost index of beam). However, more network inputs and outputs can be adopted depending on the needs of analysis, design, and computation power. Memorized trends are recalled to accomplish a final design for a given vector of attribute values (design parameters) given in an input. A vector of solutions (design results) is received in an output.

5.2.2. Reverse design

Diverse engineering scenarios including reverse designs can be explored and delved by exchanging orders of positions of input and output parameters in networks without limitations, because inputs and outputs are mapped one-to-one in ANN. Input and output parameters are switched for reverse designs. The entangled input and output relationships for a design of the doubly reinforced concrete beams are, then, memorized by an ANN when ANNs are trained on large design datasets for engineering applications. A vector of solutions (design results) is obtained for a vector of attribute values (input design parameters) by recalling memorized trends, not on the structural mechanics or knowledge. For example, reverse analysis can be presented in which beam dimensions and properties including $b$, $h$, $d$, rebar ratios, $f'_y$, $f'_c$ are determined in output side when input side has strains and target moments. Back-substitution (BS) method is developed to solve for reverse designs. Figure 4 shows procedure of BS method including two steps to solve reverse scenario which is indicated in Table 1.

Table 1 shows an example of reverse analysis with a specific goal. A methodology similar to this design example can be implemented for designs of structural components, including columns, slabs, walls, and foundations with input and output parameters which are appropriately placed.

As shown in Table 1, reverse scenario has nine input parameters shown in yellow cell of Table 1 including three reverse input parameters (factored moment ($M_w$), design strength ($\phi M_n$), and curvature ductility ($\mu_c$)) and nine output parameters shown in green cell of Table 1 including four output parameters ($h$, $d$, $p_{rn}$ and $p_{rc}$). It is noteworthy that design results are calculated accurately when single reversed input is pre-assigned without adjustments. However, adjustments will be necessary when multiple reversed inputs are pre-assigned because relationships among multiple reversed inputs may not be well established.

5.2.3. Formulation of back-substitution (BS) method

Back-substitution (BS) method consisted of reverse chained training scheme with Revised Sequence (CRS) at the first step which is proposed by Hong, Pham, and Nguyen (2021) and forward autobeam at the second step as illustrated in Figure 4. Table 2 formulate two independent networks to perform reverse designs in two steps, reverse (Step 1) based on CRS method and back-substitution for forward networks based on structural mechanics (Step 2). In reverse networks (Step 1), three reverse input parameters (factored moment $M_w$, design strength $\phi M_n$, and curvature ductility $\mu_c$) are pre-assigned in green input cell of Table 2 of first reverse network. Four reverse output parameters ($h$, $d$, $p_{rn}$, and $p_{rc}$) are, then, calculated in brown output cell of first reverse network (Step 1). In Step 2 of BS, five outputs ($\varepsilon_{rc,0.003}$, $\varepsilon_{cr,0.003}$, $\Delta_{imme}$, $\Delta_{long}$, and $C_{lb}$) are calculated in purple output cell of forward networks of Step 2 based on 10 forward inputs when four design parameters ($h$, $d$, $p_{rn}$, and $p_{rc}$) calculated in brown output cell of first reverse networks (Step 1) are back-substituted with the other six forward inputs (including beam material properties ($f'_c$, $f'_y$) and beam geometries ($b$, $L$)). Input parameters for forward networks (Step 2) are shown in ten forward pink input cell [Table 2].

Any engineering software (Autobeam in this section) can be implemented in Step 2 when forward
networks are used to find five outputs shown in purple cell for given back-substituted input design parameters shown in pink cell.

5.2.3.1. Formulation of reverse CRS of Step 1 of BS method with \( d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc} \) (Training sequence 1, TS1). In Step 1 of reverse CRS method (BS), three layers with 30 neurons are implemented to determine training sequence of CRS, resulting in training sequence with \( d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc} \) (Training sequence 1). No specific technique to find an optimum number of training parameters (including a number of epochs, hidden layers, and neurons) for best training accuracy is available (Arafa, Alqedra, and An-Najjar 2011). However, this section sought procedures to find an optimal training accuracy for each output parameter \( (h, d, \rho_{rt}, \rho_{rc}) \) based on nine training runs implementing 3 to 5 layers with 30 to 50 neurons. Training is conducted based on 100,000 datasets which is separated to training, validation, and testing data with the proportions of 70% (70,000 datasets), 15% (15,000 datasets), and 15% (15,000 datasets), respectively. Numbers of epochs and validation checks are preassigned of 50,000 and 500, respectively. The number of validation checks is applied to terminate training when the number of consecutive epochs exceeds the number of validation checks without decreasing validation performance (MSE of validation data). The best training results and ANN-based functions with sequence \( (d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc}) \) are presented in Table 3. Table 4 shows reduced final values of weight and bias (input and the first hidden layer; the fifth hidden layer and output layer) at the best epoch of training ANN on beam depth \( (d) \). These values as shown in Table 4 are the final values of weight and bias at the best epoch between input and first hidden layer. All values at intermediate epochs can be found manually, but networks have to be trained again for any intermediate epochs because Matlab only saves the values of the final epoch.

5.2.3.2. Formulation of reverse CRS of Step 1 of BS method with \( \rho_{rc} \Rightarrow \rho_{rt} \Rightarrow h \Rightarrow d \) (Training sequence 2, TS2). BS method was implemented in Table 3 using different training sequence with \( \rho_{rc} \Rightarrow \rho_{rt} \Rightarrow h \Rightarrow d \) which was obtained intuitively based on the same training parameters with three layers with 30 neurons. Training results and ANN-based functions based on training sequence determined intuitively were indicated in Table 3.

5.2.4. Design verifications based on reverse CRS of Step 1 with TS1 and TS2. In Table 2, three reverse input parameters (red input cell) including design strength \( \phi M_n \) equivalent to factored moment \( (M_u) \) and curvature ductility \( \mu_\phi \) are preassigned in green input cell, whereas four reverse output parameters \( (h, d, \rho_{rt}, \rho_{rc}) \) are calculated in brown output cell of Step 1 of BS method. Table 2 verifies accuracies of reverse designs obtained by back-substitution (BS) method with two different training sequences. Design parameters \( (h, d, \rho_{rt}, \rho_{rc}) \) calculated in brown output cell of reverse networks (Step 1) are back-substituted for input parameters of forward networks of Step 2 with the other six inputs shown in pink cell (including beam material properties \( (f_y, f_t) \) and beam geometries \( (b, l) \)). Five outputs \( (\varepsilon_{rt}, \varepsilon_{rc}, \Delta_{l_{im}}, \Delta_{l_{long}}, C_\lambda) \) have no errors when reverse design results are obtained based on Autobeam for back-substituted forward calculations of Step 2 as shown in Table 2; however, negligible error were observed for targeted parameters \( (\phi M_n, \mu_\phi) \). Any software can be implemented to perform back-substituted for forward calculation of Step 2, which is just to calculate the rest output design parameters excluding targeted design parameters which are already prescribed in input in Step 1.

Table 2 presents design accuracies of doubly reinforced concrete beams for preassigned design strength \( (\phi M_n) \) equivalent to factored moment \( (M_u) \) of 2000 and 5000 kN-m and curvature ductilities \( (\mu_\phi) \) of 6 between two different training sequences (TS1 and TS2). Design accuracies of preassigned \( \phi M_n \) equivalent to \( M_u \) of 5000 kN-m and \( \mu_\phi \) of 6 with TS2 determined intuitively were slightly better than those with TS1 determined based on three layers with 30 neurons as shown in Table 2, whereas those of \( \phi M_n \) equivalent to \( M_u \) of 5000 kN-m and \( \mu_\phi \) of 6 with TS1 were greatly better than those with TS2 as shown in Table 2. Design verifications of preassigned design strength \( (\phi M_n) \) equivalent to factored moment \( (M_u) \) from 1000 to 5000 kN-m and curvature ductilities \( (\mu_\phi) \) from 1 to 6 were performed as shown in Figure 5. In Figure 5(a), all errors of \( M_n \) for a training sequence of TS1 were acceptable (within −0.8% to 0.2%) as shown in ordinates on the right whereas those for training sequence of TS2 were insignificant (from −0.8% to 1.5%). Accuracies of \( \mu_\phi \) for a training sequence of TS1 were within −1.5% to 3.0% for \( \phi M_n \) of 5000 kN-m, whereas errors of \( \mu_\phi \) increased rapidly over than 1% momentarily at 5.2 for \( \phi M_n \) of 5000 kN-m as presented in Figure 5(b). On the other hand, errors of \( \mu_\phi \) for a training sequence of TS2 become slightly over than 1% for smaller \( \mu_\phi \) of 3.6 for \( \phi M_n \) of 5000 kN-m as presented in Figure 5(b).

6. Application of reverse design charts to design ductile beam sections

In Figure 6, an application of reverse design charts is shown with nine outputs \( (h, d, \rho_{rc}, \rho_{rt}, \varepsilon_{rt}, \varepsilon_{rc}, \Delta_{l_{im}}, \Delta_{l_{long}}, C_\lambda) \) based on nine inputs \( (\phi M_n, M_u, M_\sigma, M_0, \mu_\phi, L, b, f_y, f_t) \) as shown in Table 1. These charts were obtained based on BS method (reverse CRS with TS1 –
Figure 3. Fixed-fixed reinforced concrete beam and beam section used in ANN.

Figure 4. Procedure of BS method.

forward (Autobeam)) which was verified by Autobeam. Reverse inputs were obtained based on training sequence, (TS1: \( d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc} \)) determined based on three layers with 30 neurons, for forward CRS (Step 1) of BS. Errors were found insignificant including curvature ductility (\( \mu_{\phi} \)) and design moment strength (\( \phi M_n \)) corresponding to concrete strain of 0.003. The design charts shown in Figure 6 help engineers to be aware of design parameters, as required by codes. For example, Section 9.3.3.1, ACI 318–14 (2014) recommends that tensile rebar ratios decrease to increase rebar strains when tensile strains of rebars are less than 0.004. The design parameters meeting this requirement can be achieved by selecting design parameters corresponding to curvature ductility (\( \mu_{\phi} \)) level of 1.3 as shown in Figure 6(a) where \( L = 10,000 \) mm, \( b = 400 \) mm, \( f_y = 600 \) MPa, and \( f_c' = 30 \) MPa are preassigned. Figure 6 demonstrates how reverse design charts are applied to design tension-controlled ductile beam sections with strains greater
Table 2. Reverse design tables. (a) Factored moment ($M_u$) and design strength ($\phi M_n$) of 2000 kN·m; curvature ductility ($\mu_\phi$) of 6.0
(1) training sequence (TS1: $d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc}$). (2) training sequence (TS2: $\rho_{rc} \Rightarrow \rho_{rt} \Rightarrow h \Rightarrow d$). (b) Factored moment ($M_u$) of 5000 kN·m; curvature ductility ($\mu_\phi$) of 6.0
(1) training sequence (TS1: $d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc}$).

(a) Factored moment ($M_u$) and design strength ($\phi M_n$) of 2000 kN·m;
curvature ductility ($\mu_\phi$) of 6.0

(1) training sequence (TS1: $d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc}$)

Reverse design - BS (Reverse CRS - Forward Autobeam)

| No. | Parameter | Training results | BS | Autobeam check | Error (%) |
|-----|-----------|------------------|----|----------------|-----------|
|     |           |                  | (1)| (2)            | (3)       |
| 1   | $\phi M_u$ (kN·m) | 2000.0           | -  | 2011.8         | -0.59%    |
| 2   | $\mu_\phi$     | 6.0              | -  | 6.0            | -0.12%    |
| 3   | $M_n$ (kN·m)   | 2000.0           | -  | 2000.0         | 0.00%     |
| 4   | $M_0$ (kN·m)   | 1000             | 1000 | 1000         | -         |
| 5   | $M_t$ (kN·m)   | 500              | 500  | 500          | -         |
| 6   | $L$ (mm)       | 10000            | 10000 | 10000      | -         |
| 7   | $h$ (mm)       | 400              | 400   | 400         | -         |
| 8   | $f_r$ (MPa)    | 600              | 600   | 600        | -         |
| 9   | $f_c$ (MPa)    | 30               | 30    | 30          | -         |
| 10  | $h$ (mm)       | (4-40) 43,994 epochs; $T.M.S.E=2.17E-5, R=1.0$ | 1389 | 1389         | 1389      |
|     |               | (5-40) 15,095 epochs; $T.M.S.E=7.2E-3, R=0.988$ | 1334 | 1334         | 1334      |
| 11  | $d$ (mm)       | (5-30) 50,000 epochs; $T.M.S.E=1.29E-6, R=1.0$ | 0.006 | 0.006       | 0.006     |
|     |               | (3-50) 19,237 epochs; $T.M.S.E=2.78E-5, R=1.0$ | 0.0023 | 0.0023   | 0.0023    |
| 12  | $\rho_{rt}$   | $T.M.S.E=0.005, R=1.0$ | - | 0.00217 | -         |
| 13  | $\rho_{rc}$   | $T.M.S.E=0.005, R=1.0$ | - | 0.00218 | -         |
| 14  | $\Delta_{ml}$ (mm) | - | 2.04   | -         |
| 15  | $\Delta_{ml}$ (mm) | - | 9.19   | -         |
| 16  | $\Delta_{ls}$ (mm) | - | 89.802 | -         |

CRS sequence (TS1): $d \Rightarrow h \Rightarrow \rho_{rt} \Rightarrow \rho_{rc}$

BS method procedure

Step 1 (reverse analysis): Calculating 4 reverse outputs based on 6 ordinary inputs and 3 reverse inputs.

Step 2 (forward analysis): Calculating 5 outputs using Autobeam based on ten inputs (4 reverse output calculated from Step 1 and 6 ordinary inputs).

Autobeam check procedure

Calculating 3 outputs based on ten inputs which are identical to ten inputs of Step 2 of BS.

Error calculation

$$\text{Error} = \frac{(1) - (2)}{(1)} \times 100\%$$
than 0.005. Reverse design charts shown in Figure 6(a) were entered with a specified strain ($\varepsilon_{\text{rt, 0.003}}$) of tension reinforcement reaching 0.008 to determine curvature ductility ($\mu_\phi$) of 2.3 for a specified design moment strength ($\phi M_n$) equivalent to factored moment ($M_o$) of 2500 kN-m. The rest of design parameters for a specified design moment strength ($\phi M_n$) equivalent to factored moment ($M_o$) of 2500 kN-m can be found using reverse design charts shown in Figure 6. Compressive strain ($\varepsilon_{\text{rc, 0.003}}$) of 0.00258 for a specified design moment strength ($\phi M_n$) equivalent to factored moment ($M_o$) of 2500 kN-m is determined by Figure 6(a). Beam height ($h$) of 1240 mm, beam depth ($d$) of 1180 mm, and rebar ratios (tensile rebar ($\rho_{\text{rt}}$) of 0.01 and compressive ($\rho_{\text{rc}}$) of 0.0001) are also determined from Figure 6(b-d), respectively, when strain ($\varepsilon_{\text{rt, 0.003}}$) of tension reinforcement reaches 0.008. Both immediate of deflection ($\Delta_{\text{imme}}$) of 3.01 mm and long-term deflection ($\Delta_{\text{long}}$) of 17.6 mm of the beam are, then, determined by Figure 6(e-f), respectively, when strain ($\varepsilon_{\text{rt, 0.003}}$) of tension reinforcement reaches 0.008. Cost index of beams ($C_l$) is finally found by Figure 6(g). Final beam section is shown in Figure 6(h) where beam section (width, $b$, height, $h$, and depth, $d$), rebar ratios ($\rho_{\text{rt}}$ and $\rho_{\text{rc}}$) and locations of both tension rebars with preferred rebar diameter of
(b) Factored moment ($M_u$) of 5000 kN·m; curvature ductility ($\mu_\phi$) of 6.0

(1) training sequence (TS1: $d \Rightarrow h \Rightarrow \rho_{tr} \Rightarrow \rho_{rc}$)

| No. | Parameter | Training results | BS check | Autobeam check | Error (%) |
|-----|-----------|------------------|----------|----------------|-----------|
| 1   | $\phi M_s$ (kN·m) | 5000.0 | - | 5007.4 | -0.15% |
| 2   | $\mu_\phi$ | 6.0 | - | 5.8 | 2.73% |
| 3   | $M_s$ (kN·m) | 5000.0 | - | 5000.0 | 0.00% |
| 4   | $M_0$ (kN·m) | 2500 | 2500 | 2500 | - |
| 5   | $L$ (mm) | 1250 | 1250 | 1250 | - |
| 6   | $b$ (mm) | 400 | 400 | 400 | - |
| 7   | $f_y$ (MPa) | 600 | 600 | 600 | - |
| 8   | $f_y$ (MPa) | 30 | 30 | 30 | - |
| 9   | $T.M.S.E$ | 2.17E-5; $R$= 1.0 | 1585 | 1585 | 1585 | - |
| 10  | $d$ (mm) | 1502 | 1502 | 1502 | - |
| 11  | $\rho_{tr}$ | 0.011 | 0.011 | 0.011 | - |
| 12  | $\rho_{rc}$ | 0.0113 | 0.0113 | 0.0113 | - |
| 13  | $\rho_{tr}$ | - | 0.0222 | - | - |
| 14  | $\rho_{rc}$ | - | 0.0018 | - | - |
| 15  | $A_{lame}$ (mm) | - | 2.39 | - | - |
| 16  | $A_{long}$ (mm) | - | 11.11 | - | - |
| 17  | $CIs$ (KRW/m) | - | 168,071 | - | - |

CRS sequence (TS1): $d \Rightarrow h \Rightarrow \rho_{tr} \Rightarrow \rho_{rc}$

BS method procedure

**Step 1 (reverse analysis):** Calculating 4 reverse outputs based on 6 ordinary inputs and 3 reverse inputs.

**Step 2 (forward analysis):** Calculating 5 outputs using Autobeam based on ten inputs (4 reverse output calculated from Step 1 and 6 ordinary inputs).

Autobeam check procedure

Calculating 3 outputs based on ten inputs which are identical to ten inputs of Step 2 of BS.

**Error calculation**

$$\text{Error} = \left( \frac{(1) - (2)}{(1)} \right) \times 100\%$$

29 mm and compression rears, from which the neutral axis can be calculated are indicated. Strains are also calculated accurately as compared with calculations based on structural mechanics as shown in Table 5. Figure 6 and Table 5 are obtained based on training sequence of the order of $d$, $h$, $\rho_{tr}$, and $\rho_{rc}$ for reverse network (Step 1) when using BS method. Table 5 checks design moment strength ($\phi M_s$) equivalent to factored moment ($M_s$) of 2500 kN·m with −0.11% compared by structural mechanics solutions. Any reverse parameter to be pre-assigned is trained in input. Notably, this type of reverse technique can be implemented to control ductility of frames, helping engineers to predict performances of frames with pre-determined ductility (in terms of strains) for seismic design. Engineers can pre-assign parameters as constraining conditions of interests, not limiting these parameters to curvature ductility ($\mu_\phi$), design moment strength ($\phi M_s$), and factored moment ($M_s$) to construct reverse design charts. Design charts similar to Figure 6.
can be constructed to serve as a rapid and accurate reference, investigating a structural behavior of beam sections, and ensuring to meet code requirements. Well-balanced designs are now possible based on the proposed reverse design technologies, whereas it has been challenging to perform reverse designs based on conventional beam design. Pre-assigned parameters are not limited to curvature ductility ($\mu_\phi$), design moment strength ($\phi M_n$), and factored moment ($M_u$).

### 7. Conclusions

This study aims to develop design of doubly reinforced concrete beams based on ANN. ANNs were
Table 3. Training results of Reverse scenario of BS (Reverse CRS-Forward Autobeam) implementing 3 to 5 layers with 30, 40, and 50 neurons.

(a) TS1 sequence \( (d \Rightarrow h \Rightarrow \rho_{tr} \Rightarrow \rho_{rt}) \) determined based on 3 layers - 30 neurons.

(1) 9 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, l_1) - 1\) Output \((d)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 1   | 1,000,000 | 5      | 40      | 50,000         | 15,095                 | 15,595       | 7.20E-03    | 7.20E-03 | 0.988           |

ANN-based function: \( c = \sigma^4 \left( \sum_{i=1}^{4} W_i^c \sigma^4 \left( \sum_{j=1}^{4} W_j^i \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(2) 10 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, f_d, d^{(m)}) - 1\) Output \((h)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 2   | 1,000,000 | 4      | 40      | 50,000         | 43,994                 | 44,494       | 2.08E-05    | 2.17E-05 | 1               |

ANN-based function: \( h = \sigma^4 \left( \sum_{i=1}^{4} W_i^h \sigma^4 \left( \sum_{j=1}^{4} W_j^i \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(3) 11 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, f_d, d^{(m)}, r^{(m)}) - 1\) Output \((\rho_d)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 3   | 1,000,000 | 5      | 30      | 50,000         | 50,000                 | 50,000       | 1.20E-06    | 1.29E-06 | 1               |

ANN-based function: \( \rho_d = \sigma^4 \left( \sum_{i=1}^{4} W_i^{(m)} \sigma^4 \left( \sum_{j=1}^{4} W_j^{(m)} \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(4) 12 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, f_d, d^{(m)}, r^{(m)}, \rho^{(m)}) - 1\) Output \((\rho_p)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 4   | 1,000,000 | 3      | 50      | 50,000         | 19,237                 | 19,737       | 2.35E-05    | 2.78E-05 | 1               |

ANN-based function: \( \rho_p = \sigma^4 \left( \sum_{i=1}^{4} W_i^{(m)} \sigma^4 \left( \sum_{j=1}^{4} W_j^{(m)} \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(b) TS2 sequence \( (\rho_{tr} \Rightarrow \rho_{rt} \Rightarrow h \Rightarrow d) \) determined intuitively.

(1) 9 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, l_1) - 1\) Output \((\rho_p)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 1   | 1,000,000 | 4      | 40      | 50,000         | 9,519                  | 10,019       | 4.10E-02    | 4.26E-02 | 0.79            |

ANN-based function: \( \rho_p = \sigma^4 \left( \sum_{i=1}^{4} W_i^p \sigma^4 \left( \sum_{j=1}^{4} W_j^i \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(2) 10 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, f_d, \rho^{(m)}) - 1\) Output \((\rho_d)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 2   | 1,000,000 | 5      | 40      | 50,000         | 39,127                 | 39,627       | 1.68E-06    | 2.13E-06 | 1               |

ANN-based function: \( \rho_d = \sigma^4 \left( \sum_{i=1}^{4} W_i^{(m)} \sigma^4 \left( \sum_{j=1}^{4} W_j^{(m)} \sigma^4 \left( \sum_{k=1}^{L} \theta_{i,j,k} + b^i_k \right) \right) \right) \)

(3) 11 Inputs \((\phi M_\mu, M_\nu, M_\rho, M_\lambda, L, b, f_p, f_d, \rho^{(m)}, \rho^{(m)}) - 1\) Output \((h)\)

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|------------------------|--------------|-------------|---------|-----------------|
| 3   | 1,000,000 | 5      | 40      | 50,000         | 50,000                 | 50,000       | 2.18E-05    | 2.34E-05 | 1               |
proposed to solve structural engineering problems in general and to design ductile concrete beams in particular. ANNs were successfully implemented in designing doubly reinforced concrete beams with acceptable accuracies. A design of doubly reinforced concrete beams was attained using a memorized trend identified from structural big datasets. Independent networks were formulated to perform reverse designs in two steps, reverse CRS (Step 1) and back substitution for forward networks (Step 2). Accuracies of network design were verified based on structural mechanics.

The following conclusions can be further drawn for readers:

1. Design charts are offered for a prediction of multiple design parameters is now possible to help engineers solve reverse design problems in an accurate and rapid way. ANN-based design charts for an accurate forward and reverse designs were constructed by iterating curvature ductilities with prescribed design strength ($\phi M_{d}$) equivalent to factored moment ($M_{d}$). However, it is challenging and time-consuming for conventional softwares to generate design charts.

2. In reverse networks (Step 1), input parameters including reverse input variables are pre-assigned in input, whereas output parameters including reverse output parameters are calculated on output side. In second networks (Step 2), entire forward outputs except for targeted parameters which are already prescribed in input of Step 1 are calculated in output of forward networks of Step 2. Design parameters calculated in output of first reverse networks (Step 1) are back-substituted for forward networks of Step 2 with other inputs.

3. Step 1 of BS is to obtain four brown reverse outputs ($h$, $d$, $\rho_{tr}$, and $\rho_{sc}$) for given three green reverse inputs ($\phi M_{tr}$, $M_{d}$, and $\mu_{d}$), which are shown in Table 2, respectively, where design moment strength ($\phi M_{d}$) equals to factored moment $M_{d}$ of 2000 and 5000 kN-m were prescribed in input. In Table 3, three layers

| Table 3. (Continued) |

| (4) 12 Inputs ($\phi M_{tr}$, $M_{d}$, $\rho_{tr}$, $L$, $b$, $f_{tr}$, $P_{tr}$, $P_{sc}$, $h_{(a)}$) – 1 Output (d) |

| No. | Data | Layers | Neurons | Required Epoch | Best Epoch for Training | Stopped Epoch | Training MSE | Test MSE | # at Best Epoch |
|-----|------|--------|---------|----------------|-------------------------|---------------|-------------|----------|----------------|
| 4   | 1,00,000 | 4      | 40      | 50,000         | 50,000                  | 50,000        | 1.99E-06    | 1.96E-06 | 1              |

ANN-based function $h = \sigma^{1}(W_{(0\times4)} \sigma^{2}(W_{(40\times40)} \sigma^{3}(W_{(40\times40)} \mathbf{x} + b_{1}^{(40)} + b_{2}^{(40)}) + b_{1}^{(40)}) + b_{2}^{(40)})$.

Table 4. Final values of weights and biases at the best epoch of training ANN on beam depth (d) following TS1.

(a) Matrix of weight and bias values between input layer and the first hidden layer.

| $W_{(40\times9)}$ | $b_{(10\times1)}$ |
|------------------|------------------|
| 0.829            | −0.603           |
| 1.001            | 0.577            |
| −1.386           | −0.603           |
| −0.493           | −0.333           |
| ...              | ...              |
| 1.532            | −0.606           |
| 0.099            | −0.320           |
| −1.885           | 0.802            |
| −0.520           | 1.012            |
| ...              | ...              |
| 0.371            | 0.241            |
| ...              | ...              |
| 0.291            | 0.357            |
| 0.638            | 0.638            |

(b) Matrix of weight and bias values between the $S^{th}$ hidden layer and the output layer.

| $W_{(1\times40)}$ | $b$ |
|------------------|-----|
| [−0.018]         |     |
| [0.155]          |     |
| [0.371]          |     |
| ...              |     |
| [0.371]          |     |
| [1.3255]         |     |
| [0.291]          |     |
| [0.357]          |     |
| 0.638            |     |
with 30 neurons are used to determine training sequence (TS1) of Step 1, with training sequence based on $d \Rightarrow h \Rightarrow \rho_{tr} \Rightarrow \rho_{cr}$.

(4) Reverse design results of doubly reinforced concrete beams based on proposed networks developed in this study were found close to those designed by engineers. Errors were found insignificant including for curvature ductility ($\mu_{\phi}$) and design moment strength ($\phi M_n$) when using BS (Reverse CRS – Forward Autobeam) method.

(5) Even well-trained networks cannot yield accurate design results with unexpected errors when input parameters that do not represent real engineering behaviors are implemented. Engineers should be questioning the quality of input design parameters when this happens.

Figure 5. Verification of design moment strengths and ductility of BS method. (a) Design strength ($\phi M_n$). (b) Curvature ductility ($\mu_{\phi}$).
(6) Simplistic, robust, and fast but accurate design results with well-balanced structures are now available to engineers and project planners in estimating their project perspectives by using the proposed technique.

(7) ANNs can offer a rapid design of doubly reinforced beams according to design codes, enabling reverse analysis and design. Many applications can be developed in various areas by using the method proposed in this study, offering a novel way of extending engineers’ imagination that is implausible with conventional design methods.

(8) The ANN can perform some limited extrapolation with reverse input parameters being selected appropriately with even when design parameters lay outside the datasets. The ANN showed good extrapolation when input conflicts were avoided. However, the network accuracy

Figure 5. Continued.
Figure 6. Use of reverse design charts to design ductile beam sections corresponding to strain (\(\varepsilon_{rt,0.003}\)) of tension reinforcement of 0.008 and \(\phi M_u = M_u (2500 \text{ kN} \cdot \text{m})\) selected in (a).

(a) Determining curvature ductility and compressive strain for targeted tensile strain (0.008) and \(\phi M_u = M_u (2500 \text{ kN} \cdot \text{m})\) selected in (a).

(b) Beam height (h) for curvature ductility (strain (\(\varepsilon_{rt,0.003}\))) of tension reinforcement of 0.008 and \(\phi M_u = M_u (2500 \text{ kN} \cdot \text{m})\) selected in (a).

Figure 6. Use of reverse design charts to design ductile beam sections corresponding to strain (\(\varepsilon_{rt,0.003}\)) of tension reinforcement of 0.008; CRS sequence of (TS1: d ⇒ h ⇒ \(\rho_{rt}\) ⇒ \(\rho_{rc}\)).
(c) Beam depth \(d\) for curvature ductility (strain \(\varepsilon_{\text{cr},0.003}\) of tension reinforcement of 0.008) and \(\phi M_k = M_k(2500\,\text{kN\cdot m})\) selected in (a)

(d) Tensile and compressive rebar ratios \(\rho_{\text{t}}\) and \(\rho_{\text{c}}\) for curvature ductility (strain \(\varepsilon_{\text{cr},0.003}\) of tension reinforcement of 0.008) and \(\phi M_k = M_k(2500\,\text{kN\cdot m})\) selected in (a)

Figure 6. Continued.
(c) Immediate deflections ($\Delta_{\text{immediate}}$) for curvature ductility (strain ($\varepsilon_{t,0.001}$) of tension reinforcement of 0.008) and $\phi M_n = M_n$ (2500 kN·m) selected in (a)

(f) Long-term deflections ($\Delta_{\text{long}}$) for curvature ductility (strain ($\varepsilon_{t,0.003}$) of tension reinforcement of 0.008) and $\phi M_n = M_n$ (2500 kN·m) selected in (a)

Figure 6. Continued.
(g) Determining cost index of beam \((CI_b)\) for curvature ductility (strain \((\varepsilon_{rt,0.003})\) of tension reinforcement of 0.008) and \(\phi M_s = M_s (2500 \text{ kN·m})\) selected in (a)

(h) Final beam section corresponding to strain \((\varepsilon_{rt,0.003})\) of tension reinforcement of 0.008
Table 5. Reverse design table to design ductile beam sections corresponding to strain \((\varepsilon_{rt,0.003})\) of tension reinforcement of 0.008; design verification of Figure 6.

| No. | Parameter               | BS   | Autobeam check | Error (%) |
|-----|-------------------------|------|----------------|-----------|
| 1   | \(\phi M_n\) (kN·m)    | 2500.0 | 2502.9         | -0.11%    |
| 2   | \(\mu \phi\)           | 2.3  | 2.3            | -0.13%    |
| 3   | \(M_0\) (kN·m)         | 2500.0 | 2500.0         | 0.00%     |
| 4   | \(M_0\) (kN·m)         | 1250  | 1250.0         | 0.00%     |
| 5   | \(M_L\) (kN·m)         | 625   | 625.0          | 0.00%     |
| 6   | \(L\) (mm)             | 10000 | 10000.0        | 0.00%     |
| 7   | \(b\) (mm)             | 400   | 400.0          | 0.00%     |
| 8   | \(f'_c\) (MPa)         | 600   | 600            | 0.00%     |
| 9   | \(f_c\) (MPa)          | 30    | 30             | 0.00%     |
| 10  | \(h\) (mm)             | 1240  | 1240           | 0.00%     |
| 11  | \(d\) (mm)             | 1180  | 1180           | 0.00%     |
| 12  | \(\rho_{rt}\)          | 0.010 | 0.010          | 0.00%     |
| 13  | \(\varepsilon_{rc}\)   | 0.0001 | 0.0001         | 0.00%     |
| 14  | \(\varepsilon_{rt,0.003}\) | 0.0080 | 0.0078         | 2.50%     |
| 15  | \(\varepsilon_{rc,0.003}\) | 0.00258 | 0.00260     | -0.62%    |
| 16  | \(\Delta l_{mme}\) (mm) | 3.01   | 3.02          | -0.43%    |
| 17  | \(\Delta l_{long}\) (mm) | 17.60   | 17.60         | 0.00%     |
| 18  | \(C_Ib\) (KRW/m)       | 95.600 | 95.608        | -0.01%    |

CRS sequence (TS1): \(d \rightarrow h \rightarrow \rho_{rt} \rightarrow \rho_c\)

Procedure to apply design charts:

Step 1: Determining nine parameters based on preassigned nine parameters using design charts.

Step 2: Verifying results obtained from design charts by Autobeam based on ten forward inputs.

Error calculation:

\[
(3) = \frac{(1)-(2)}{(1)} \times 100\%
\]
became weak in general when values too far outside the training datasets were used. This is because input conflicts were often caused with weak extrapolation capability, which resulted in significant errors when reverse input parameters were inappropriately selected.

**Nomenclature**

| Symbol | Description |
|--------|-------------|
| L      | Beam length |
| h      | Beam height |
| d      | Effective beam depth |
| b      | Beam width |
| \( f_c \) | Compressive concrete strength |
| \( f_y \) | Yield rebar strength |
| \( \rho_{st} \) | Tensile rebar ratio |
| \( \rho_{rc} \) | Compressive rebar ratio |
| \( M_0 \) | Moment due to dead load |
| \( M_l \) | Moment due to live load |
| \( \phi M_0 \) | Design strength excluding moment due to self-weight load |
| \( M_y \) | Moment due to self-weight load |
| \( M_y(1.2M_0+1.6M_l) \) | Beam moment considering effects of dead load and live load |
| \( \mu_p \) | Curvature ductility |
| \( \varepsilon_{rc,0.003} \) | Compressive rebar strain at concrete strain of 0.003 |
| \( \Delta_{imme} \) | Immediate deflection due to live load, \( M_l \) |
| \( \Delta_{long} \) | Sum of time-dependent deflection due to sustained loads and immediate deflection due to any additional live load |
| \( C_l \) | Cost index of beam |
| BS     | Back-substitution method |
| CRS    | Chained training scheme with Revised Sequence |
| TS     | Training sequence, for example, TS1 is training sequence 1 |
| \( T.MSE \) | Mean square errors of testing data |
| R      | Regression value |

**Disclosure statement**

No potential conflict of interest was reported by the author(s).

**Funding**

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government [MSIT 2019R1A2C2004965].

**References**

Abambres, M., and E. O. Lantsoght. 2020. “Neural Network-based Formula for Shear Capacity Prediction of One-way Slabs under Concentrated Loads.” *Engineering Structures* 211: 110501. doi:10.1016/j.engstruct.2020.110501

Allahyari, H., I. M. Nikbin, S. Rahimi, and A. Heidarpour. 2018. “A New Approach to Determine Strength of Perfobond Rib Shear Connector in Steel-concrete Composite Structures by Employing Neural Network.” *Engineering Structures* 157: 235–249. doi:10.1016/j.engstruct.2017.12.007.

Anderson, D., E. L. Hines, S. J. Arthur, and E. L. Eiap. 1997. “Application of Artificial Neural Networks to the Prediction of Minor Axis Steel Connections.” *Computers & Structures* 63 (4): 685–692. doi:10.1016/S0045-7949(96)00080-6.

Arefa, M., M. Alqeda, and H. An-Najjar. 2011. “Neural Network Models for Predicting Shear Strength of Reinforced Normal and High Strength Concrete Deep Beams.” *Journal of Applied Sciences* 11 (2): 266–274. doi:10.3923/jas.2011.266.274.

Armaghani, D. J., G. D. Hatzigeorgiou, C. Karamani, A. Skentou, I. Zoumpoulaki, and P. G. Asteris. 2019. “Soft Computing-based Techniques for Concrete Beams Shear Strength.” *Procedia Structural Integrity* 17: 924–933. doi:10.1016/j.prostr.2019.08.123.

Armaghani, D. J., and P. G. Asteris. 2020. “A Comparative Study of ANN and ANFIS Models for the Prediction of Cement-based Mortar Materials Compressive Strength.” *Neural Computing and Applications* 1–20. doi:10.1007/so0521-020-05244-4.

Asteris, P. G., D. J. Armaghani, G. D. Hatzigeorgiou, C. G. Karayannis, and K. Pilakoutas. 2019. “Predicting the Shear Strength of Reinforced Concrete Beams Using Artificial Neural Networks.” *Computers and Concrete* 24 (5): 469–488. doi:10.12989/cac.2019.24.5.469.

Asteris, P. G., and V. G. Mokos. 2020. “Concrete Compressive Strength Using Artificial Neural Networks.” *Neural Computing and Applications* 1–20. doi:10.1007/so0521-019-04663-2.

Azim, I., J. Yang, M. F. Javed, M. F. Iqbal, Z. Mahmood, F. Wang, and Q. F. Liu. 2020. “Prediction Model for Compressive Arch Action Capacity of RC Frame Structures under Column Removal Scenario Using Gene Expression Programming.” *In Structures* 25: 212–228. doi:10.1016/j.istruc.2020.02.028.

Bka, M. A. R., C. Ngamkhanong, Y. Wu, and S. Kaewrunruen. 2021. “Recycled Aggregates Concrete Compressive Strength Prediction Using Artificial Neural Networks (Anns).” *Infrasctructures* 6 (2): 17. doi:10.3390/infrasctructures6020017.

Charalampakis, A. E., and V. K. Papanikolakou. 2021. “Machine Learning Design of R/C Columns.” *Engineering Structures* 226: 111412. doi:10.1016/j.engstruct.2020.111412.

Chen, X. L., J. P. Fu, J. L. Yao, and J. F. Gan. 2018. “Prediction of Shear Strength for Squat RC Walls Using a Hybrid ANN–PSO Model.” *Engineering with Computers* 34 (2): 367–383. doi:10.1016/s0376-0177(15)30047-5.

Dahou, Z., Z. M. Sbatai, A. Castel, and F. Ghomari. 2009. “Artificial Neural Network Model for Steel–concrete Bond Prediction.” *Engineering Structures* 31 (8): 1724–1733. doi:10.1016/j.engstruct.2009.02.010.

Das, S., and S. Choudhury. 2019. “Influence of Effective Stiffness on the Performance of RC Frame Buildings Designed Using Displacement-based Method and Evaluation of Column Effective Stiffness Using ANN.” *Engineering Structures* 197: 109354. doi:10.1016/j.engstruct.2019.109354.

Flood, I., and N. Kartam. 1994a. “Neural Networks in Civil Engineering. I: Principles and Understanding.” *Journal of Computing in Civil Engineering* 8 (2): 131–148. doi:10.1061/(ASCE)0887-3801(1994)8:2(131).
Flood, I., and N. Kartam. 1994b. “Neural Networks in Civil Engineering. II: Systems and Application.” Journal of Computing in Civil Engineering 8 (2): 149–162. doi:10.1061/(ASCE)0887-3801(1994)8:2(149).

Gupta, T., and R. K. Sharma. 2011. “Structural Analysis and Design of Buildings Using Neural Network: A Review.” International Journal of Engineering and Management Sciences 2 (4): 216–220.

Hong, W. K., T. D. Pham, and V. T. Nguyen. 2021. “Feature Selection Based Reverse Design of Doubly Reinforced Concrete Beams.” Journal of Asian Architecture and Building Engineering. doi: 10.1080/13467581.2021.1928510

Huang, X., J. Zhang, J. Sresakooolchai, and S. Kaewunruen. 2021. “Machine Learning Aided Design and Prediction of Environmentally Friendly Rubberised Concrete.” Sustainability 13 (4): 1691. doi:10.3390/su13041691.

Lee, S., and C. Lee. 2014. “Prediction of Shear Strength of FRP-reinforced Concrete Flexural Members without Stirrups Using Artificial Neural Networks.” Engineering Structures 61: 99–112. doi:10.1016/j.engstruct.2014.01.001.

Lee, S., J. Ha, M. Zokhirova, H. Moon, and J. Lee. 2018. “Background Information of Deep Learning for Structural Engineering.” Archives of Computational Methods in Engineering 25 (1): 121–129. doi:10.1007/s11831-017-9237-0.

Lu, S., M. Koopalipoor, P. G. Asteris, M. Bahri, and D. J. Armaghani. 2020. “A Novel Feature Selection Approach Based on Tree Models for Evaluating the Punching Shear Capacity of Steel Fiber-reinforced Concrete Flat Slabs.” Materials 13 (17): 3902. doi:10.3390/ma13173902

Morfidis, K., and K. Kostinakis. 2018. “Approaches to the Rapid Seismic Damage Prediction of R/C Buildings Using Artificial Neural Networks.” Engineering Structures 165: 120–141. doi:10.1016/j.engstruct.2018.03.028.

Neves, A. C., L. Gonzalez, J. Leander, and R. Karoumi. 2017. “Structural Health Monitoring of Bridges: A Model-free ANN-based Approach to Damage Detection.” Journal of Civil Structural Health Monitoring 7 (5): 689–702. doi:10.1007/s13349-017-0252-5.

Nguyen, D. H., and W. K. Hong. 2019. “Part I: The Analytical Model Predicting Post-Yield Behavior of Concrete-Encased Steel Beams considering Various Confinement Effects by Transverse Reinforcements and Steels.” Materials 12 (14): 2302. doi:10.3390/ma12142302.

Oh, B. K., K. J. Kim, Y. Kim, H. S. Park, and H. Adeli. 2017. “Evolutionary Learning Based Sustainable Strain Sensing Model for Structural Health Monitoring of High-rise Buildings.” Applied Soft Computing 58: 576–585. doi:10.1016/j.asoc.2017.05.029.

Rafiei, M. H., and H. Adeli. 2017. “A Novel Machine Learning-based Algorithm to Detect Damage in High-rise Building Structures.” The Structural Design of Tall and Special Buildings 26 (18): e1400. doi:10.1002/tal.1400.

Sharifi, Y., F. Lotfi, and A. Moghboli. 2019. “Compressive Strength Prediction Using the ANN Method for FRP Confined Rectangular Concrete Columns.” Journal of Rehabilitation in Civil Engineering 7 (4): 134–153. doi:10.22075/JRCE.2018.14362.1260.

Standard, A. A. 2014. “Building Code Requirements for Structural Concrete.” (ACI 318–14). In American Concrete Institute. The MathWorks. 2020b. Deep Learning Toolbox™ Getting Started Guide. MathWorks. Natick, Massachusetts, United State. Retrieved from https://www.mathworks.com/help/deeplearning/getting-started-with-deep-learning-toolbox.html

Yaseen, Z. M., M. T. Tran, S. Kim, T. Bakhshpoori, and R. C. Deo. 2018. “Shear Strength Prediction of Steel Fiber Reinforced Concrete Beam Using Hybrid Intelligence Models: A New Approach.” Engineering Structures 177: 244–255. doi:10.1016/j.engstruct.2018.09.074.

Zarba, S. E. H. A. M., M. Norouzi, R. Allemand, V. Hunt, A. Helmicki, and C. Venkatish. 2018. “Vibration-based Cable Condition Assessment: A Novel Application of Neural Networks.” Engineering Structures 177: 291–305. doi:10.1016/j.engstruct.2018.09.060.