Grey Wolf Optimizer-Based ANN to Predict Compressive Strength of AFRP-Confined Concrete Cylinders

Amir Hasanzade-Inallu

Department of Earthquake Engineering, Science and Research Branch, Islamic Azad University, Daneshgah sq., Sattari highway, Tehran (Postal code: 1477893855), Iran
Corresponding Email: amirhasanzade@outlook.com

Abstract
One of the well-known and attractive solutions for strengthening of reinforced concrete columns are Fiber-Reinforced Polymer (FRP) confinements. However, most of the models developed in literature are based on a general equation originally given for steel confinements. Because of the material differences between FRPs and steel, we aimed to model the compressive strength of aramid fiber-reinforced polymer confined concrete cylinders without any assumption on the form of the model. To be useful, Artificial Neural Networks (ANNs) must be trained using an optimization algorithm. In this study, we used a recently proposed optimization algorithm named Grey Wolf Optimizer for training. After developing ANN models, we compared them with five existing equations. The statistical parameters indicated the superiority of the proposed ANN models over existing ones.

Keywords:
Aramid Fiber-Reinforced Polymer (AFRP)
Concrete
Artificial Neural Networks (ANNs)
Grey Wolf Optimizer (GWO)

1. Introduction

Reinforced concrete columns need repair and/or strengthening when they are damaged under external loads, e.g., earthquake, or due to corrosion of steel. Also, in the case of change in structural use, strengthening may be necessary. A well-known solution to this problem is the use of external confinements, which provide additional load bearing capacity and also increase the ductility of the columns. External confinements come in many forms such as circular and spiral reinforcements, steel jacketing, concrete jacketing, and fiber reinforced polymer (FRP) jacketing. Because of their lower weight-to-strength ratio, resistance to corrosion, and adding minimal area to the original structure, FRP confinements are an excellent substitute for concrete and steel jacketing [1,2]. Common types of FRP confinement include carbon fiber (CFRP), glass fiber (GFRP), and aramid fibers (AFRP) [3]. Among these FRPs, the demand for AFRPs is expected to be more than 120 thousand tons by the year 2020; however, it is less treated in the literature and there are few stress-strain models specific for them [4].

Because of their desirable characteristics, researchers have conducted many studies on estimating the axial strength and stress-strain relationship of FRP-confined concrete columns [5–9] and the result has been the development of over 90 stress-strain models [10]. Also, now there is a wide availability of guidelines for the design of FRP-confined reinforced concrete columns [7].

Most of the stress-strain models proposed in literature are based on the general equation proposed by Richart et al. [11] which was originally developed to estimate the steel jacket confinements. Recent studies have concluded that for FRP confinements, this equation overestimates and therefore is not the most reliable model for FRP-confined concrete columns [5,12,13]. Artificial Neural Networks (ANNs) have been successfully leveraged in many problems in
Engineering domain [14–17]. In this study, we aimed to estimate the compressive strength of AFRP-confined concrete cylinders by introducing a model trained by the recently proposed Grey Wolf Optimizer (GWO) algorithm. Section 2 gives an overview of ANNs, section 3 describes the implementation of GWO to train an ANN, and finally section 4 gives results and provides discussion.

2. Artificial Neural Networks

Artificial Neural Networks (ANNs) were designed to model the way the human brain performs tasks. They are usually implemented as a software program or as hardware components. ANNs consist of information processing units, called neurons, which are connected to each other [18].

Multilayer Perceptrons (MLPs) are a class of artificial neural networks that consist of one input layer, one or more hidden layers, and one output layer. In a fully-connected ANN, each neuron in a given layer is connected to other neurons in layers before and after the given layer. The function of a single neuron is to first multiply its inputs by weights, which show the importance of a particular input on output, then add a constant, called bias, to the result, and then apply a function, called activation or transfer function, to the resulting sum. One of the common activation functions for regression problems is the hyperbolic tangent function [19,20].

Initial weights of an ANN are usually randomly assigned, and therefore the output of the network will differ from the target values of the output. To minimize the error of an ANN, i.e., the difference between network output and target values, the weights and biases of the network must be adjusted in a process called training [18–20].

The training process is an optimization process and the methods for solving this optimization problem can be divided into two categories: gradient-based and metaheuristic methods. One of the reasons for choosing metaheuristic methods over gradient-based methods is that, unlike gradient-based methods, metaheuristic methods do not get stuck in local minima. The answer given by metaheuristic methods is not necessarily the global minimum; however, these methods are usually designed to explore and exploit a large portion of solution space [18–20].

3. Methods and Materials

3.1. Parameters influencing compressive strength

To select parameters influencing the compressive strength of FRP-confined concrete cylinders, we used the parameters employed in models published in literature. Input parameters considered were: specimen diameter (D), unconfined concrete strength (f_{uc}), AFRP modulus of elasticity (E_{fr}), ultimate tensile strength of AFRP (f_{fr}), and the total thickness of AFRP (t_{fr}). Ultimate axial stress of AFRP-confined concrete cylinder (f_{uc}) was considered to be the target value to be modelled [4].

3.2. Experimental Data

The database of experimental results was obtained from Djafar-Henni and Kassoul [4] paper. It includes 81 experimental results for AFRP-confined concrete cylinders collected from literature. The unconfined concrete strength of specimens, ranged from 20 MPa to 128 MPa. Specimens did not have internal steel or FRP reinforcements. All the specimens were confined with wet layup method, and the confinements were continuous. To eliminate the effect of specimen slenderness, only specimens having height-to-diameter ratio less than three were chosen. The specimens selected, all had failed due to AFRP rupture at the ultimate condition [4].

To train a network data were divided into two groups of training and testing. 70% (57 samples) of data were selected to be in training group and the remaining (24 samples) were used as testing data. The reason for having these two groups is that we do not want the network to overemphasize the experimental data and to be able to generalize for data not encountered during training. This is a known phenomenon called overfitting, in which the training is so focused on training data and minimizing the error on all data that the network loses its generalization capabilities. The division of data between these two groups was done randomly [19,20].

Scaling is always useful in modelling. The goal of this process is to set the data into a specified (usually small) range. Without scaling, training algorithm might converge slowly or even diverge [19]. We scaled each parameter, i.e., five inputs and one output, using the following relation:

\[ Y_{sc} = \frac{2(Y - Y_{min})}{Y_{max} - Y_{min}} - 1 \]  (1)
where $Y_{sc}$ is the scaled value of a given parameter, $Y_{max}$ is the maximum value of the parameter, $Y_{min}$ is the minimum value of the parameter, and $Y$ is the un-scaled value of the parameter [21].

The statistics of experimental data are given in Table 1.

| Table 1. Statistics of experimental data (81 cylinders). |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | D (mm) | H (mm) | $f_{c0}$ (MPa) | $E_f$ (GPa) | $f_t$ (MPa) | $t_f$ (mm) | $f_{cc}$ (MPa) | $f_{cc}/f_{c0}$ |
| Avg | 130.38 | 305.57 | 61.67 | 113.67 | 2427.02 | 0.66 | 106.85 | 1.84 |
| SD | 38.62 | 105.35 | 31.56 | 23.24 | 726.39 | 0.74 | 45.23 | 0.52 |
| CoV | 0.30 | 0.34 | 0.51 | 0.20 | 0.30 | 1.12 | 0.42 | 0.28 |
| Min | 63.00 | 126.00 | 23.10 | 13.60 | 128.50 | 5.04 | 35.16 | 1.09 |
| Max | 194.00 | 582.00 | 128.00 | 128.50 | 3732.00 | 4.98 | 204.51 | 2.30 |
| Range | 131.00 | 456.00 | 104.90 | 114.90 | 3502.00 | 4.98 | 169.35 | 2.21 |

Note: $D$ = specimen diameter; $H$ = specimen height; $f_{c0}$ = unconfined concrete strength; $E_f$ = AFRP modulus of elasticity; $f_t$ = AFRP ultimate tensile strength; $t_f$ = thickness of AFRP; $f_{cc}$ = AFRP-confined ultimate axial strength; $f_{cc}/f_{c0}$ = strength enhancement

The frequency distribution of ultimate axial strength of AFRP-confined concrete cylinders is shown in Figure 1.

3.3. Optimal ANN Selection

The topology, i.e., the number of neurons and hidden layers, of an artificial neural network is problem-dependent. To find the optimum ANN topology we used the maximum number of neurons of 14 and trained the network using all the topologies that can be formed with 1 to 14 neurons in 1 to 3 hidden layers. For example, for one hidden layer networks, we trained networks with 1 to 14 neurons in hidden layer, for two hidden layer networks, we kept number of neurons in first hidden layer at 1 and changed the number of neurons in second hidden layer from 1 to 13, and then selected 2 neurons in first hidden layer and 1 to 12 neurons in second layer and so on. We trained a total of 105 different topologies, and the last trained topology was 12 neurons in first hidden layer, 1 in second hidden layer, and 1 in third hidden layer. The error metric used to select the most accurate ANN among trained networks was Mean Squared Error (MSE). [19,20,22].
3.4. Grey Wolf Optimizer

To perform optimization, one of the recently proposed metaheuristic algorithms named the Grey Wolf Optimizer (GWO) was used. GWO is inspired by the social structure and hunting technique of grey wolves. Grey wolves have a very strict social dominant hierarchy. The leaders of the pack are called alphas, which are responsible for making decisions. Next in the hierarchy are the betas, which are subordinate wolves that help the alpha in decision-making or other pack activities. Delta wolves have to submit to alphas and betas. They are usually the scouts, sentinels, elders, hunters, and caretakers. The lowest ranking grey wolves are the omegas, which have to submit to all the other wolves [23].

The algorithm works by first initializing the population of wolves, called search agents, and then defining alpha, beta, and delta wolves as the best, second best, and third best solutions according to their cost. These wolves estimate the location of the prey, i.e., an optimal solution. The remaining search agents update their position with respect to alpha, beta, and delta wolves, whilst adding a random movement to explore the search space. Half of the iterations of GWO are dedicated to exploration and therefore avoiding local minima, and the other half are dedicated to exploitation of the optimum solution. The algorithm iterates for a predefined number of times and finally the position of the alpha, which is position of the lowest cost solution found, is returned [23].

We developed a software program in the MATLAB 2018b [24] software to first scale the data, and then for each of the topologies specified divide data into training and testing groups. The program then trained the selected topology using GWO algorithm and calculated the statistics of the ANN for training and testing data. The results for each of the topologies were saved and after the training, the top five networks with the lowest value of MSE on testing data were selected. The number of search agents was set to 100 and the number of iterations was chosen to be 500. All the networks had hyperbolic tangent function as the activation function of the hidden layers and linear $y = x$ function as the activation function of the output layer.

4. Results and Discussion

4.1. Top five ANN models

After training 105 different topologies, their statistics were calculated and top five artificial neural networks were selected based on their test data MSE being low. The reason for selecting the MSE of test data as the metric for choosing best performing network is that, the networks were not trained using test data and their metrics on test data indicate the networks’ ability to generalize on data not encountered before. Network names are designated by ANN-GWO mL($n_1$-$n_2$-$n_3$), where m designates the number of hidden layers, and $n_1$, $n_2$, and $n_3$ show the number of neurons in first, second, and third hidden layers respectively. Table 2 lists the statistics of the top five ANNs selected.

<table>
<thead>
<tr>
<th>Network</th>
<th>MSE (TRAIN)</th>
<th>MSE (TEST)</th>
<th>R (TRAIN)</th>
<th>R (TEST)</th>
<th>RMSE (TRAIN)</th>
<th>RMSE (TEST)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN-GWO 3L (2-10-2)</td>
<td>0.007</td>
<td>0.005</td>
<td>0.986</td>
<td>0.994</td>
<td>0.085</td>
<td>0.067</td>
</tr>
<tr>
<td>ANN-GWO 3L (2-6-6)</td>
<td>0.037</td>
<td>0.006</td>
<td>0.944</td>
<td>0.991</td>
<td>0.193</td>
<td>0.075</td>
</tr>
<tr>
<td>ANN-GWO 3L (3-7-4)</td>
<td>0.035</td>
<td>0.008</td>
<td>0.937</td>
<td>0.990</td>
<td>0.187</td>
<td>0.087</td>
</tr>
<tr>
<td>ANN-GWO 3L (5-5-4)</td>
<td>0.003</td>
<td>0.011</td>
<td>0.995</td>
<td>0.982</td>
<td>0.054</td>
<td>0.107</td>
</tr>
<tr>
<td>ANN-GWO 3L (10-2-2)</td>
<td>0.007</td>
<td>0.012</td>
<td>0.989</td>
<td>0.986</td>
<td>0.084</td>
<td>0.108</td>
</tr>
</tbody>
</table>

4.2. Existing Models

To compare the results of our models with those published in the literature, we selected five models namely Bisby and Ranger [7], Djafar and Kassoul [4], Wei and Wu [25], Pham and Hadi [26], and Wu et al. [27]. Mean Squared Error (MSE) was calculated for five existing models and five artificial neural networks given in Table 2, using Eq. (2).

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (o_i - p_i)^2$$

where n is the number of samples, $o_i$ is the $i$th observed value, and $p_i$ is the $i$th predicted value. The calculated values of MSE on all 81 samples for models is depicted in Figure 2 in the order of increasing values of MSE. As shown in Figure 2, three ANN models ANN-GWO 3L (2-10-2), ANN-GWO 3L (5-5-4), and ANN-GWO 3L (10-2-2)
outperform the rest of the models. The MSE values for these networks are 0.0124, 0.0125, and 0.0229 respectively.

![Mean Squared Error](image)

**Figure 2.** Mean squared error values for ANN models and models published in literature

For the ten models considered, the Root Mean Squared Error (RMSE) was calculated using Eq. (3).

\[
RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (o_i - p_i)^2}
\]  

(3)

![RMSE](image)

**Figure 3.** Root mean squared error values for ANN models and models published in literature
where \( n \) is the number of samples, \( o_i \) is the \( i \)th observed value, and \( p_i \) is the \( i \)th predicted value. The calculated values of \( \text{RMSE} \) on all samples for the models is depicted in Figure 3 in the order of increasing values of \( \text{RMSE} \). As shown in Figure 3, three ANN models ANN-GWO 3L (2-10-2), ANN-GWO 3L (5-5-4), and ANN-GWO 3L (10-2-2) outperform the rest of the models. The \( \text{RMSE} \) values for these networks are 0.1115, 0.1118, and 0.1513 respectively.

To compare the fitness of ten models considered, the coefficient of determination \( (R^2) \) was calculated using Eq. (4).

\[
R^2(O,P) = \left( \frac{\sum(o_i - \bar{o})(p_i - \bar{p})}{\sqrt{\sum(o_i - \bar{o})^2 \sum(p_i - \bar{p})^2}} \right)^2
\]

(4)

where the calculation is done for \( n \) samples, and \( o_i \) is the \( i \)th observed value, and \( p_i \) is the \( i \)th predicted value. The calculated values of coefficient of determination \( (R^2) \) on all 81 samples for the ten models is depicted in Figure 4 in the order of decreasing values of \( R^2 \). As shown in Figure 4, three ANN models ANN-GWO 3L (2-10-2), ANN-GWO 3L (5-5-4), and ANN-GWO 3L (10-2-2) outperform the rest of the models. The \( R^2 \) values for these networks are 0.9937, 0.9936, and 0.9884 respectively.

![Figure 4. Coefficient of determination \((R^2)\) values for ANN models and models published in literature](image)

### 4.3. Top model selection

By comparing the results of MSE, RMSE, and \( R^2 \), it’s apparent that the performance of ANN-GWO 3L (2-10-2) is slightly better than the rest. So, this network was chosen for further analysis. Figure 5 depicts the experimental values of AFRP-confined concrete strength vs. their predicted value by ANN-GWO 3L (2-10-2). The comparison between the experimental and predicted values of AFRP-confined concrete axial strength for individual concrete cylinder specimen is depicted in Figure 6.

This model is a 4-layer artificial neural network with 2 neurons in first hidden layer, 10 neurons in second hidden layer, 2 neurons in third hidden layer, and 1 neuron in output layer corresponding to output of the network. The activation (transfer) functions of hidden layers are hyperbolic tangent, and the activation function of the output layer is the linear function \( y = x \).

### 4.4. Top ANN model weights and biases

Because under the normal circumstances, a user will not have access to the computer file of the trained ANN-GWO 3L (2-10-2) model, its weights and biases were extracted from the trained model. Before using, input data must be scaled using Eq. (1), by substituting max and min values from Table 1 for each of the five inputs.
The input is denoted by a 5x1 vector called $\mathbf{a}^{(1)}$. To calculate FRP-confined compressive strength, $f_{cc}^{\text{predict}}$, Eqs. (5) - (9) must be used.
\[ a^{(2)} = \tanh(\theta^{(1)} \times a^{(1)} + b_1) \]  
(5)

\[ a^{(3)} = \tanh(\theta^{(2)} \times a^{(2)} + b_2) \]  
(6)

\[ a^{(4)} = \tanh(\theta^{(3)} \times a^{(3)} + b_3) \]  
(7)

\[ f_{cc}^{\text{predict (scaled)}} = \theta^{(4)} \times a^{(4)} + b_4 \]  
(8)

\[ f_{cc}^{\text{predict}} = \frac{f_{cc}^{\text{predict (scaled)}} + 1}{2} \times (f_{cc,\text{max}} - f_{cc,\text{min}}) + f_{cc,\text{min}} \]  
(9)

where \( f_{cc,\text{max}} \) and \( f_{cc,\text{min}} \) are the max and min values of \( f_{cc} \) given in Table 1 and

\[ \theta^{(1)} = \begin{bmatrix} -0.2172 & -0.1782 & -1.7085 & -0.6468 & 0.5180 \\ -0.2811 & 0.6175 & 2.0811 & 1.6469 & 2.5624 \\ 1.9844 & 3.4537 \\ 0.9288 & -3.7324 \\ 0.0822 & 4.8500 \\ 3.8774 & 1.9254 \end{bmatrix} \]

\[ \theta^{(2)} = \begin{bmatrix} -3.6720 & -3.2537 \\ 3.5972 & -2.8924 \\ 3.7198 & 2.4226 \\ -3.6963 & 3.0429 \\ 2.9036 & -2.8123 \\ 4.4013 & -0.5004 \end{bmatrix} \]

\[ \theta^{(3)} = \begin{bmatrix} 1.0192 & 0.9722 & 0.4890 & -1.2845 & -0.4179 & -1.3026 & -0.4560 & -0.9630 & -0.5784 & 0.1434 \\ 1.2451 & -0.7525 & 0.9938 & 1.3165 & 0.4768 & -0.2543 & -0.1680 & -1.3714 & 0.2489 & 0.1612 \end{bmatrix} \]

\[ \theta^{(4)} = \begin{bmatrix} -0.1949 & 1.0152 \end{bmatrix} \]

\[ b_1 = \begin{bmatrix} 2.2592 \\ 0.2181 \end{bmatrix} \quad b_2 = \begin{bmatrix} -4.9494 \\ -3.1657 \\ -2.6992 \\ -1.8829 \\ -0.0355 \\ 0.5456 \\ 1.4053 \\ -2.5140 \\ 3.9393 \\ 4.4221 \end{bmatrix} \quad b_3 = \begin{bmatrix} 1.6760 \\ 1.0965 \end{bmatrix} \quad b_4 = -0.2758 \]
5. Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

6. References


