Optimization of Mixture Proportions of Roller Compacted Concrete Based on Neural Network Modeling

Ali Akbar Ramezanianpour[i], Khosro Sadeghi Marzaleh[ii], Mohammad Esmaeilnia Omran[iii], Hamid Rahmani[iv]

ABSTRACT

Roller compacted concrete (RCC) is a no slump concrete which is widely used in the construction of dams and road pavements. Mixture design of such concretes containing minimum cement pastes is necessary to control the thermal problems in mass concretes. With an accurate model for prediction of RCC compressive strength, it is possible to optimize RCC mixture design. Nevertheless, RCC is a highly complex material that modeling its behavior is a difficult task and needs nonlinear modeling methods.

In the present study, attempt was made to propose a neural network model for prediction of compressive strength of roller compacted concretes. Necessary data obtained from the mixture designs of various laboratory and field test results. Neural network (NN) has strong capability of modeling complex, multi-variable and nonlinear problems such as RCC mixture design. In the concrete mixtures, minimum cement pastes are used to design concretes for the required compressive strength. With the proposed NN model in this investigation, it is possible to estimate the compressive strength of roller compacted concretes as the output of the model.

KEYWORDS

Roller compacted concrete, Compressive strength, Minimum RCC pastes, Mixture design, Neural network.

1. INTRODUCTION

Roller Compacted Concrete is a no slump concrete which is widely used in the construction of dams and road pavements. Considering economical benefits and fast speed of construction in the RCC dams, there is rapid expansion of this method throughout the world. An important issue in RCC dam constructions is the mixture design optimization. Today, RCC dam engineers use standard codes and recommended methods to design mixture proportion. In these methods, many limitations and assumptions are made. Thus, in practice, trial and error procedure is used and some samples of RCC mixes with various proportions are made to achieve the required specification. In addition, for controlling the thermal problems, reducing cost and optimizing mixture, using different type of pozzolans and using new admixture, other parameters are considered in the mix design and make it more complicated. In fact, designing and optimizing RCC mixtures require huge effort in sample preparation. However, by creating models to predict the compressive strength of RCC concrete mixtures, time and cost for designing a concrete mixture proportions are reduced. Due to the complex behavior and nonlinearity in the RCC concretes, nonlinear modeling methods must be used to achieve the model with appropriate accuracy that can applied to optimize RCC mixtures. One of the most powerful methods in nonlinear modeling considering all effective parameters in parallel is Neural Network (NN) modeling method. The following characteristics of backpropagation neural networks, which have been adequately described in the literature [1, 2], make them very attractive and appropriate for predicting concrete properties.

1. They can establish mapping between inputs (i.e.,

[i] A. Ramazanianpour, Professor, Amirkabir University of Technology, Civil Engineering Department, Tehran, Iran (e-mail: aaramce@aut.ac.ir).
[ii] KH. Sadegi, M.Sc student, Amirkabir University of Technology, Civil Engineering Department, Tehran, Iran (e-mail: mskhosro@yahoo.com).
[iii] M. Esmaeilnia O. Assistant Professor, Kordestan University, Civil Engineering Department, Sanandaj, Iran (Tel: 021-22259346)
iv H. Rahmani, Ph.D student, Amirkabir University of Technology, Civil Engineering Department, Tehran, Iran (e-mail: hrahmani@cic.aut.ac.ir).

proportions of concrete constituents) and outputs (i.e., concrete properties) based on historical data, without any explicit model being declared, even if those mappings are highly non-linear. This is especially useful when additives and admixtures are combined with the basic concrete constituents [3].

2. They are able to tolerate errors in the training data because of their generalizing activity and concrete mix design data very often has 'error' in the form of high variability [3].

3. They can be updated based in successive trial mixes, as retraining of neural network is very easy [3].

4. Their primary weakness of lacking a theory-based model is mitigated by the general accepted fact that mix design is essentially an empirical process [3].

There are different backpropagation neural networks. In this paper, the Multi-Layer Perceptron architecture, which is the most common method for engineering applications and can map the inputs and outputs [4], was used. This type of neural networks is used to model different concrete mixture design. Yeh [5], Kasperkiewics [6], Lai [7] and Lee [8] applied the NN for predicting properties of conventional concrete and high performance concrete. Bai [9] developed neural network models that provide effective predictive capability in respect to the workability of concrete incorporating metakaolin and fly ash. Ji-Zong [10] developed an automatic knowledge-acquisition system based on neural networks to design concrete mixes. Guang and zong [11] proposed a method to predict 28-day compressive strength of concrete by using multi-layer feed forward neural networks. Dias and Pooliyadda [12] used back propagation neural networks to predict the strength and slump of ready mixed concrete and high strength concrete. Also, Ahmet Oztas [13] developed a NN model for prediction in parallel compressive strength and slump of high strength concrete with two hidden layers. The aim of this paper is to present a NN model that can predict the compressive strength of RCC with high accuracy and optimize the RCC mixture proportion.

2. Multi-Layer Perceptron Neural Network

The Multi-Layer Perceptron (MLP) is the most common neural networks for engineering applications and can learn any continuous functions with arbitrary accuracy [14] if suitable numbers of hidden layer neuron are applied in. A Multi-Layer Perceptron (MLP) is a type of feed-forward neural network that has the strong capabilities of learning and nonlinear processing and processing tolerance characteristics of inaccuracy and uncertainty and robustness and back propagation is used for updating the weights of each layer and bias of each neuron in the layers based on the error at the network output [10]. As shown in Figure 1, MLP always has at least three layers, the input layer, the hidden layer, and the output layer [15].

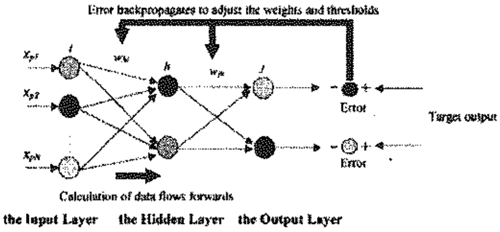

Figure 1. The architecture of a BP neural network.

More than one hidden layer can be used for some applications. The presence of these hidden layers allows the network to present and compute more complicated associations between patterns [15]. x_{p1}, x_{p2}, … , x_{pN} are the N components of input vector X_p, and W_{hi} and W_{jh} are the connection weights between nodes of different layers. The nodes (neurons) of neighboring layers are fully connected. A back propagation network functions on the basis of a large number of neurons. A neuron is an information-processing element. Neurons in the input layer just transfer the input data to the hidden layer, with no calculation happening. While in the hidden layer and the output layer, a neuron acts as seen in Figure 2 [10]. Function f in the figure names as "activation function" and b_j is the bias on neuron j. The tangent sigmoid and sigmoid (Figure 3) are commonly used in MLP as activation function in hidden layers and also the linear function is used in the output layer.

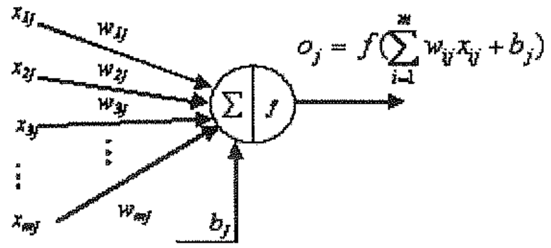

Figure 2.Neuron and its action.

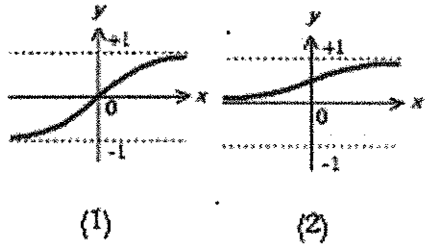

Figure 3. Activation function of neuron; (1) Tangent sigmoid, (2) Sigmoid.

The MLP implementation involves two processes. The first process involves the presenting the patterns (samples or data) to input layer and the calculation of the data following forward form the input layer to the output layer. The second process involves the calculation of error between network output and target pattern (Eq .1) and propagation error signals backward from the output layer to the input layer and adjusting the connection weights and neurons bias to minimize the error function (Eq. 1).

$$MSE = \frac{1}{N.S_o} \sum_{j=1}^{S_o} \sum_{i=1}^{N} (t_{ij} - o_{ij})^2 \qquad (1)$$

where MSE (Mean Square Error) is the cost function, t_{ij} and o_{ij} are the desired output and network output in the output layer neuron i and pattern jth of data respectively, and N is the number of output layer neuron and So is the number of pattern.

3. NN MODEL CONSTRUCTION AND DATA COLLECTION

In order to attain accurate model to use it in optimization process, different network parameters and several network architectures were applied and examined. Database that is used for modeling and development of NN model will is described in the next part.

3.1. Data collection

According to RCC mix design main effective parameters, the authors collected the necessary data from Jaghin and Zirdan RCC dams built in Iran. The collected data consist of 116 RCC laboratory samples. Three types of mineral additive materials were used; Khash, Trass (Two Iranian natural pozzolans) and Indian fly ash with pozzolanic activity of 73, 78.4 and 78.7 percentages, respectively.

3.2. NN model architecture and configurations

As described above, there are different kinds of NN. But the multi-layer Perceptron architecture, which is the most common NN for engineering application, was developed in this research. The developed NN has three layers. In the input layer, there are 11 neurons corresponding to 11 effective RCC factors that were dosage of cement, dosage of pozzolan, dosage of water, dosage of sand, dosage of gravel, maximum size of aggregate, crushed sand-total sand ratio, crushed gravel-total gravel ratio, type of sample compaction, pozzolanic activity and water reducing admixture content. The RCC sample compaction type is a qualitative value, and needs to be changed into quantitative value. Values 0 and 1 were used to quantify Modified Proctor and VeBe compaction type, respectively. In the output layer, there is one neuron corresponding to 91-day RCC compressive strength.

The number of hidden layer neurons affects the model accuracy and model generalization, computing time of the learning process and learning the rule of the training set [14]. If only few neurons are presented in the hidden layer of NN model, it is not possible to learn the rule of the learning set. When there are too many neurons in the hidden layer, however, the model fit data points in the learning set too well. It learns, in addition to the rule of the data set, the noise in the learning set, and generalization thus becomes worse [14].

In the present research work, to create model with acceptable accuracy and generalization, the number of hidden layer neurons was selected as variable in the range of 1 to 16 neurons. Two ordinary, tangent sigmoid and sigmoid, activation functions are employed for hidden layer neurons in the experiments independently and only linear transfer function used in the output layer neuron as activation function. In order to get good generalization, a method called early stopping has been used as the stopping criterion of learning process. In this technique, the available data is divided into three subsets. The first subset is the training set, which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process. The validation error will normally decrease during the initial phase of training, as does the training set error. However, when the network begins to over fit the data, the error on the validation set will typically begin to rise, whereas the training error still keeps decreasing. At this condition, the training is stopped and the network is considered to be the solution. The test set (3rd set of the division of the data) error is not used during the training, but it is used to compare different models. With this method, it is possible to train the network that has smoothly regularization the data points or the patterns of RCC mixtures and the model shows suitable generalization. Using early stopping method, the data in the training set, testing set, and the validation set should have similar characteristics [14]. Therefore, approximately 23 present of the data was selected showing similar trend in the cumulative frequency of 91-day compressive strength with the entire data (see Figure 4).

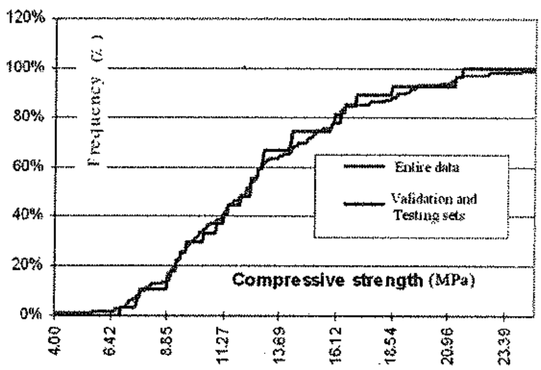

Figure 4. Cumulative distribution function for data sets.

Then, approximately 50 percent of the selected data was randomly reselected as the validation set and the rest of the data was used as testing set. In this way, the 116 data are divided into three groups, 89 set as the training data, 14 set as the validation data, and 13 set as the testing data. The Training algorithm employed here was the Levenberg-Marquardt (LM) back propagation algorithm with batch-update, seemed to be one of the fastest training algorithm among all the gradient decent derived back propagation algorithms [14]. It is important to note that with LM algorithm, it is possible to decrease the hidden layer neurons. Also, each configuration of the network is trained with 10 simulations, each with different starting condition (random initial weights and biases).

3.3. Data Pre-processing

Data scaling is another essential step for network training. One of the reasons for pre-processing the output data is the application of a sigmoidal activation function within the network [16]. Outputs limits of tangent sigmoid and sigmoid functions are [-1,1] and [0,1] respectively (see Figure 3). Scaling of the inputs in the range [-1,+1] greatly improves the learning speed, as these values fall in the region of the sigmoidal activation function where the output is most sensitive to variation of the input values. Another reason is that the components which form an input vector have different quantitative limits, so data normalization is needed [10]. It is therefore recommended to normalize the input and output data before presenting them to the network. There are different methods of linear transformations to normalize the data. The two transformation functions used in this work are shown in Eq.(2) and Eq.(3). The first equation scales data so that they fall in the range [-1,+1] and the second one scales data so that they will have the mean and standard deviation values equal to zero and one, respectively.

$$x_{pi} = 2\left(\frac{x_i - x_{\min}}{x_{\max} - x_{\min}}\right) - 1 \qquad (2)$$

where x_i and x_{pi} are the *i*th pattern of each input or output parameters after and before normalization, respectively, and x_{min} and x_{max} are the maximum and minimum values of data for each input or output parameters.

$$x_{pi} = \frac{x_i - x_{mean}}{x_{std}} \qquad (3)$$

where x_{mean} and x_{std} are the mean and standard deviation values of data for each input or output parameters. The normalized data sets after transformation with Eq.(2) and Eq.(3) and original data without transformation were presented independently to the network for training purposes and reaching to the suitable model.

4. Results of NN Model Construction

In this research, a program was written in the Matlab software with the help of its Neural Network Toolbox to train and test the MLP neural network with different parameters described in the previous section. After several processing the network with the best performance on both training set and the testing set was selected as the final neural network model. The proposed NN model with 7 neurons in the hidden layer was found using the Eq. (2) as transformation function of pre-processing the data.

The performance of training set, validation set and testing set can be seen in Figure 5. The results in Figure 5 indicate that the neural network model is successful in learning the relationship between the different input parameters and the output (compressive strength). Results of testing set in Figure 5 also show that the neural network is capable of generalizing between input variables and output with reasonably a good prediction. The correlation coefficient (*R*) between the network respond and the target and mean squared error (*MSE*) for compressive strength values in training set, validation set and testing set are given in Table 1. It is clearly seen (Table 1 and Figure 5) that the proposed NN model is appropriate for prediction of 91 day compressive strength of RCC mixtures.

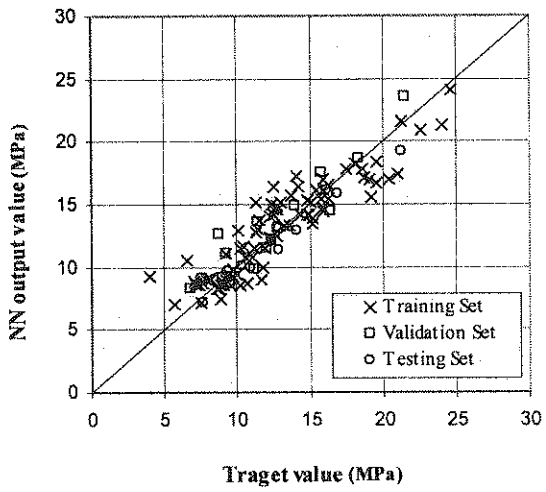

Fig .5. The performance of training, testing and validation sets of mix-design data.

TABLE 1
STATISTICAL VALUE OF PROPOSED MODEL

	Training Set	Validation Set	Testing set
R	0.9402	0.9067	0.9516
MSE	0.0334	0.2300	0.2461

5. Optimization of Mixture Proportion

It is important to control the temperature of RCC dams. Post cooling system cannot be used in such dams. The best way to reduce the temperature in RCC dams is the design of concrete mixtures with low cement content. Less cement in the concrete mixtures led to low heat and low cost concrete. In this investigation, attempt was made to find RCC mixtures proportions containing minimum cement pastes for 11 levels of compressive strength (from 10 to 20 MPa) by using the proposed NN model. The mixtures contain Khash pozzolan which was used in Jagin RCC dam. The limitations in the model are as follows:

- Concrete mixtures contain no admixtures.

- The maximum pozzolan-binder ratio was 0.4.
- The maximum water-binder ratio was 0.7.
- Effective air content of the mixture was 1.5%.
- Natural aggregate is used with 25.4 mm MSA.
- The specific gravity of pozzolan cement and aggregates are 2.56, 3.16 and 2.7, respectively.

It is easy to estimate the minimum cement required for the mixture using NN model. The proposed model acts as function that its input parameters are the independent variables and its output is the dependent variable. The Matlab Optimization Toolbox includes routines for many types of optimization. One of them is the constrained nonlinear minimization. A Matlab program was written to minimize the binder content in the mixtures. In this program the proposed NN model consists of a function with its input and output variables and constrained equation, and the goal function of the minimization is defined as the sum of cement and pozzolan contents. The result of the mixture optimization is seen in Table 2. The presented mixture proportions have minimum cement pastes according to required compressive strength and other proposed constraints. This kind of optimization can be extended to cost minimization.

TABLE 2
OPTIMIZED RCC MIXTURES PROPORTIONS

No.	Concrete ingredients in kg/m^3							91-day Compressive Strength (MPa)
	Cement (kg)	Pozzolan (kg)	Binder (kg)	Water (kg)	MSA (mm)	Sand (kg)	Gravel (kg)	
1	58.42	38.95	97.37	68.16	61.55	1100.0	1284.5	9.7
2	60.21	40.00	100.21	69.69	59.05	1094.6	1283.2	11.1
3	60.00	40.00	100.00	68.46	55.25	1098.9	1282.3	11.9
4	60.00	40.00	100.00	69.88	50.68	1097.1	1280.2	13.0
5	60.00	40.00	100.00	69.20	45.52	1100.0	1279.2	14.2
6	60.00	40.00	100.00	68.26	38.71	1100.0	1281.7	14.9
7	60.00	40.00	100.00	29.48	25.4	1021.4	1357.1	15.8
8	64.96	43.30	108.27	75.79	25.4	1100.0	1253.7	17.3
9	76.63	51.09	127.73	79.30	25.4	1100.0	1226.0	18.1
10	134.12	89.41	223.53	73.02	25.4	1100.0	1153.5	19.0
11	199.31	132.87	332.18	77.20	53.5	740.61	1400.0	20.0

6. Conclusions

In this paper, the NN model was developed for predicting the RCC 91-day compressive strength. The proposed model performance and its accuracy and generalization show that the use of NN model will provide a useful decision-making tool for RCC dam engineers. The model can present a new method for RCC mixture proportion designing that is more suitable and accurate than the traditional methods. The model will save time, decrease wasting materials and reduce the design cost. The model is useful to study the effect of each variation of the mixture components on compressive strength of RCC concretes. With the help of the proposed model and ordinary minimization method, RCC mixtures containing minimum cement pastes were designed for the required compressive strengths. The simplicity of optimization of RCC mixture shows that NN modeling has strong capability of modeling nonlinear, parallel and multi-variable problems.

7. References

[1] Flood I., Kartam N., "Neural networks in civil engineering – I: Principles and understanding", *ASCE journal of computing in civil engineering*, April 1994, 8(2), pp. 131-148.

[2] Carpenter WC., Barthelemy J.F., "Common misconceptions about neural networks as approximators", *ASCE journal of computing in civil engineering*, July 1994, 8(3), pp. 345-358.

[3] Dias W.P.S., Pooliyadda S.P., "Neural network for predicting properties of concretes with admixtures", *Construction and Building Materials*, 2001, Vol.15, pp. 371-379.

[4] Marta Sebastia, Inaki Fernandez Olmo, Angle Iranien, "Neural network prediction of unconfined compressive strength of coal fly ash –cement mixtures", *Cement and Concrete Research*, 2003, 33, pp. 137-146.

[5] Yeh I.C., "Modeling of strength of high-performance concrete using artificial neural networks", *Cement and Concrete Research*, 1998, Vol. 28, pp. 1797-1808.

[6] Kasperkiewics J., Racz j., Dubrawski A., "High performance concrete strength prediction using artificial neural network", *ASCE journal of computing in civil engineering*, 1995, 9(4), pp. 131-148.

[7] Lai S., Sera M., "Concrete strength prediction by means of neural network", *Construction and Building Materials*, 1997, 11(2), pp. 93-98.

[8] Lee SC., "Prediction of concrete strength using artificial neural networks", *Engineering Structure*, 2003, 25, pp. 849-857.

[9] Bai J., Wild S., Ware JA., Sabir BB., "Using neural networks to predict workability of concrete incorporating metakaolin and fly ash", *Advance in engineering software*, 2003, 34, pp. 663-669.

[10] Ji-Zong W., Hong-Guang N., Jin-Jun H., "The application of automatic acquisition of knowledge to mix design of concrete", *Cement and Concrete Research*, 1999, 29, pp. 1875-80.
[11] Guang NH., Zong WJ., "Prediction of compressive strength of concrete by neural networks", *Cement and Concrete Research*, 2000, 30, pp.1245-1250.
[12] Dias WPS., Pooliyadda SP., "Neural networks for predicting properties of concrete with admixtures", *Construction and Building Materials*, 2001, 15, pp. 371-379.
[13] Ahmet Öztas, Murat Pala, Erdogan Özbay, Erdogan Kanca, Naci Caglar, M. Asghar Bhatti., "Prediction the compressive strength and slump of high strength concrete using neural networks", *Construction and Building Materials*, 2005, IN PRESS.
[14] Liqun Ren, Zhiye Zhao, "An optimal neural network and concrete strength modeling", *Advances in engineering software*, 2002, 33, pp. 117-130.
[15] Ragip Ince, "Prediction of fracture parameters of concrete by Artificial Neural Networks", *Engineering Fracture Mechanics*, 2004, 71, pp. 2143–2159.
[16] Rafiq M.Y., Bugmann G., Easterbrook D.J., "Neural network design for engineering applications", *Computers & Structures*, 2001, 79, pp. 1541-1552.