American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

http://asrjetsjournal.org/

Application of Neural Networks and Adaptive-Network-Based Fuzzy System in the Prediction of Optimum Bitumen Content for Asphaltic Concrete Mixtures

Moussa. S. Elbisy[a*], M. H. Alawi[b], M. A. Saif[c]

[a,b,c]*Department of Civil Engineering, Umm Al-Qura University, Makkah, Saudi Arabia*

[a]*Email: mselbisy@uqu.edu.sa*

[b]*Email: masaif@uqu.edu.sa*

[c]*Email: cenmha@yahoo.com*

Abstract

The objective of this study is to explore the applicability of artificial neural networks (ANNs) and Adaptive-Network-Based fuzzy System (ANFIS) for predicting the bitumen content (OBC) of asphaltic concrete mixtures based on the experimental data. Samples were collected from different regions in Makkah region in Saudi Arabia during construction and tested at laboratories of Umm Al-Qura University for bitumen content, gradation of aggregate determination. Asphaltic concrete mixtures data were used to test the performance of the ANNs and ANFIS models. Among the two ANN models (a feed-forward back propagation (BP) and a radial basis function (RBF)) employed for this investigation, the BP neural network was found to be superior to RBF network for prediction of the OBC of asphaltic concrete mixtures. For improving model prediction efficiency, optimization of network structure and spread are important for BP and RBF types of the network, respectively. A BPNN model having a structure 3-8-4-1 (three neurons in input and eight neurons in first hidden layers, four neurons in second hidden layer and one neuron in output layer) produced better prediction performance efficiencies with an accuracy of 96.37%. The BPNN (3-8-4-1) model was fairly close to the corresponding actual values of OBC with the average error of 1.1854% and 1.01% for trained and tested data respectively. The results of the testing of ANFIS were indicated almost same performance of the BPNN (3-8-4-1) model.

Keywords: Bitumen content; artificial neural networks; Adaptive-Network-Based fuzzy System; prediction.

----

* Corresponding author

1. Introduction

In order to design a road superstructure, two main tasks have to be accomplished, namely, the mix design of the asphalt concrete to be used for each of the layers of the pavement and the thickness design of the pavement itself. Focusing the attention on the mix design task, currently, all over the world, experimental procedures carried out in a road laboratory are adopted [1-3]. The laboratory tests used to evaluate the physical properties and the mechanical resistance of components and mixtures are quite time consuming; moreover, skilled laboratory technicians have to be involved [4]. The optimum bitumen content of asphaltic concrete mixture (OBC) is greatly influenced by several parameters: Aggregate (type- size- shape- roughness- angularity – texture – gradation - absorption), Filler (type – shape – size - quantity), and bitumen (type – penetration - viscosity). Consequently, developing the optimum bitumen content of asphaltic concrete mixture requires an extensive understanding of the relation between these parameters and the properties of the resulting matrix. Determination of OBC is an essential component for design of asphaltic concrete mixture. There are many empirical formulas [5] for determination of OBC has been developed. In this study, the applications of artificial neural networks and Adaptive-Network- Based FIS (ANUS) [6], which is a combination of ANN and FIS to predict the OBC of asphaltic concrete mixtures have been investigated. Most of the modern research in modeling asphaltic concrete mixtures aims to construct mathematical models to describe the relationship between components of mixtures behavior. These models consist of mathematical rules and expressions that capture these varied and complex behaviors. Moreover, an asphaltic concrete mixture is a highly nonlinear material, so the modeling of its behavior, in particular its stability, is difficult and time-consuming task. Recent advances in research in the area of model identification have revealed approaches for inducing models from data, based on learning systems. These approaches can determine the relationships between the input, and output variables from data presented to them, without resorting to describe these relationships explicitly in mathematical form. One of the learning systems, which are the subject of this study, is the artificial neural networks. Artificial Neural networks (ANNs) have the ability to recognize the hidden pattern in the data and accordingly estimate the values. Provision of model-free solutions, data error tolerance, built in dynamism and lack of any exogenous input requirement makes the network attractive. A neural network is an information processing system modeled on the structure of the human brain. Its merit is the ability to deal information whose interrelation is ambiguous or whose functional relation is not clear. One of the advantages of ANNs compared to traditional regression models is that they do not require a prior regression model, which relates input and output data and in general is difficult because these models are not known[7]. In the past years. Fuzzy Inference System (FIS), which is based on expertise expressed in terms of 'IF-THEN' rules, has been employed in different subjects. FIS can be used to predict uncertain systems and its application does not require knowledge of the underlying physical process as a precondition. To the best knowledge of the authors, this paper presents the first application of FIS coupled with ANNs to prediction OBC of asphaltic concrete mixture estimation based on the experimental data.

2. Method and material used

2.1 Data acquisition

The main objective of this study is developing a model to predict the OBC of asphaltic concrete mixture. For this aim, at first it is needed to prepare data and construct data base for training and testing the neural network model. Loose mixtures, which were used in this research, were collected from different regions in Makkah area during the construction of new roads and tested at the asphalt laboratory of Umm Al-Qura University for bitumen content, gradation of aggregate determination. Compaction for 75 blows each side at 150oc temperature was performed.

2.2 Material properties

The aggregate gradation of all bituminous mixtures lies within the upper and lower limits gradation of grade A wearing course for heavy traffic (Table 1), as well as stability, flow, voids, VMA, VFB according to the specifications of the Ministry of Transportation (M.O.T) specification (Table 2).

Table 1: M.O.T. specifications for Gradation of wearing course, class A.

Sieve Size Designation	M.O.T. specification limits (% Passing)
¾"	100
½"	80 – 95
3/8"	-
# 4	48 – 62
# 10	32 – 45
# 40	16 – 26
# 200	4 – 8

2.3 Neural network models

ANNs are flexible computing frameworks for modeling a broad range of nonlinear problems. One significant advantage of the ANN models over other classes of nonlinear model is that ANNs are universal approximators which can approximate a large class of functions with a high degree of accuracy. Their power comes from the parallel processing of the information from the data. No prior assumption of the model form is required in the model building process. Instead, the network model is largely determined by the characteristics of the data.

Table 2: M.O.T in Saudi Arabia design criteria (review form MRDWS 410D.c) for wearing course class A of heavy traffic.

Mix properties	Criteria
Compaction (No. of blows each end of specimen)	75
Marshall Stability, kg (min)	1000
Marshall Flow (mm)	2.0-3.5
Loss of Marshall Stability, % (Max.)	25
Voids in mix, %	4.0-7.0
Marshall mixing temperature, ºC	155-165
Marshall Compaction temperature, ºC	140-150
Voids in mineral aggregate, min	15
Voids filled with bitumen, %	65-75

Two well-known ANN models: the feed-forward BP, and the radial basis function (RBF) were employed as preliminary exploratory models to investigate which would be more suitable for the purpose of prediction of LWC. The MATLAB Toolbox (MathWorks, Inc., Natwick, MA) was used to create BP, and RBF type neural networks for this study.

2.3.1 Back propagation neural network (BPNN)

Within the class of feed forward neural network paradigms, the most widely used neural network paradigm is the back propagation. The back propagation neural network (BPNN) is preferred over other neural network paradigms because of its simplicity and ease in implementation [8]. As a feed forward architecture, back propagation models contain an input layer, an output layer and at least one hidden layer, which are all fully interconnected. Although back propagation models embody feed forward architectures, where information is passed in one direction, the models actually implement multi directional operations. Back propagation utilizes supervised learning, which requires a desired output to be declared during the training phase. During the training phase, root mean square error (RMSE) is calculated between the desired output and the actual output. The RMSE is then propagated backwards to the input layer and the connection weighs between the layers are readjusted. After the weighs have been adjusted and the hidden layer neurons have generated an output result, the error value is again re-determined. Before the training phase begins the total number of input neurons and the number of hidden layer neurons and the total number of iteration (propagations) must be declared. When the training phase initializes, the connection weights between the input and hidden layers are assigned random values by means of an activation function. The goal of any training algorithm is to minimize the global (mean sum squared) error *E*. The BP algorithm calculates the error, is then used to adjust the weights first in the output layer, and then distributes it backward from the output to hidden and input nodes (Fig. 1). This is done using the steepest gradient descent principle where the change in weight is directed towards negative of the error gradient, i.e.

$$\Delta w_n = \alpha \Delta w_{n-1} - \eta \frac{\partial E}{\partial w} \tag{1}$$

Where w is the weight between any two nodes; Δw_n, Δw_{n-1} are the changes in this weight at n and n-1 iteration, α the momentum factor, and η is the learning rate.

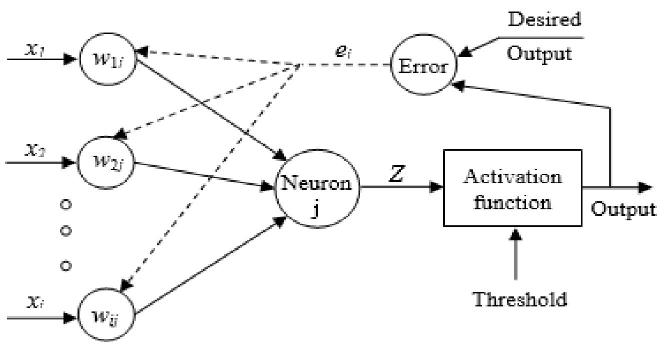


Figure 1: Neuron weight adjustments.

Little research has been conducted to find good initial weights [9, 10]. The initial weights are randomly generated between -1 and 1 with a random number generator (MathWorks, Inc., Natwick, MA). The value of the learning parameter is not fixed. The optimization of learning parameter is highly problem dependent and should be selected so that oscillation in error surface can be avoided [9]. The function of the hidden layer nodes is to detect relationships between network inputs and outputs. If there is insufficient number of hidden nodes, it may be difficult to obtain convergence during training time, as the network may be unable to create adequately complex decision boundaries. On the other hand, if too many hidden nodes are used, the network may lose its ability to generalize. In addition, keeping the number of hidden layer nodes to a minimum reduces the number of weights that need to be adjusted, and hence reduces the computational time needed for training.

2.3.2 Radial basis function (RBF) neural network

The RBF network consists of an input layer, one hidden layer of basis functions or neurons and an output layer with feed-forward architecture. The input layer is composed of *n* input nodes. The hidden layer consists of *J* locally tuned units and each unit has radial basis function acting like a hidden node. The input nodes are fully connected to the hidden layer nodes. Connections between the input and hidden layer have unit weights and, as a result, do not have to be trained [11]. The neurons in the hidden layer do not use the weighted sum of inputs and sigmoid transfer function, which are typical in BPNN. Instead, the outputs of the hidden layer neurons, each of which represents a basis function, are determined by the Euclidean distance between the network input and the center of the basis function. As the input moves away from a given center, the neuron output drops off rapidly to zero [12]. Figure 2 shows a schematic diagram of RBF network.

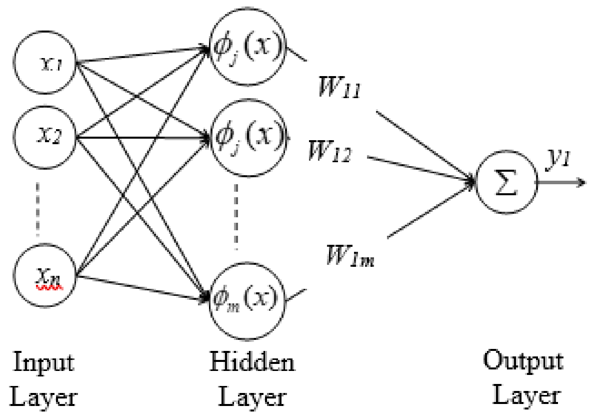


Figure 2: Schematic diagram of a RBF network.

RBF networks are able to provide a local representation of an *N*-dimensional space. This is made by restricted influence zone of the basis functions. The parameters of this basis function are given by a reference vector (core or prototype) μ_j and the dimension of the influence field σ_j. The response of the basis function depends on the Euclidian distance between the input vector X, $X = [x_1, x_2, \dots, x_n]$, and the prototype vector μ_j, and depends also on the size of influence field:

$$\varphi_j(x) = exp\left(-\frac{\|x-\mu_j\|^2}{2\sigma_j^2}\right) \quad (2)$$

The RBF network output is formed by a weighted sum of the hidden layer neuron outputs and the unity bias. The output of the network is computed by the equation:

$$y = \sum_{j=1}^{m} w_j \varphi_j(x) \quad (3)$$

Where W_j , connection weight between j[th] hidden neuron (of n number) and output neuron.

The 'newrb' function available in the commercial MATLAB Toolbox was used to create a radial basis neural network. Initially there is no radial basis neuron. It iteratively creates one radial basis neuron at a time and adds neuron to the network until either the sum squared error falls beneath an error goal (MSE here) or the maximum number of neurons is reached. The maximum number of neurons depends upon the width (spread) of radial basis function. If the spread is larger, the slope of a radial basis function gets smoother that leads to a large area around input vector and several neurons may respond to an input vector. Therefore, if the spread is small, the radial basis function is very steep so that neurons with the weight closest to the input will have a larger output than other neurons.

2.4 ANFIS model

An Adaptive-Network-Based Fuzzy Inference System (ANFIS) [6] is a Sugeno type FIS in which the problem of fine-tuning membership functions of premise variables is carried out by a feed-forward neural network. ANFIS combines the advantages of both neural networks (e.g. learning capabilities, optimization capabilities, and connectionist structures) and fuzzy inference systems (e.g. human like 'IF-THEN' rule thinking and ease of incorporating expert knowledge). The basic idea behind these neuro-adaptive learning techniques is very simple. They provide a methodology for the fuzzy modeling procedure to learn information about a data set, in order to compute the membership function parameters that best allow the associated FIS to track the given input-output data. ANFIS is based on the premise of mapping a FIS into a neural network structure so that the membership functions and consequent part parameters are optimized using a hybrid learning algorithm. In this algorithm, parameters of the membership functions are determined by a neural network back-propagation learning algorithm while the consequent parameters by the least square method. Fig. 3 shows the structure of ANFIS including two inputs x, y, and one output and two rules. The first step is the fuzzifying layer in which A_i, and B_i, are the linguistic labels. The output of this layer is the membership functions of these linguistic labels. In other words, in this step, the premise parameters are calculated. The second step calculates the firing strength for each rule. The output of this step is the algebraic product of the input signals. The third step is the normalized layer. Every node in this layer calculates the ratio of the *i*th rule's firing strength to the sum of all rule's firing strength. The fifth layer computes the overall output as the summation of all incoming signals, which represents the results of OBC.

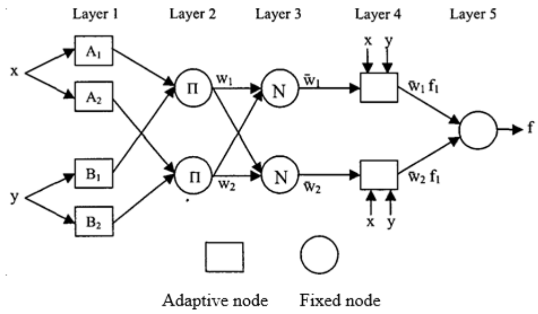


Figure 3: ANFIS architecture.

3. Analysis of results and discussion

To statistical compression of predicted and observed of the OBC of asphaltic concrete mixtures, the root mean square error *RMSE*, the coefficient of efficiency E_f, and the correlation coefficient *R*, were computed using the following expressions:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i - O_i)^2} \tag{4}$$

$$R = \frac{\sum_{i=1}^{N}\left(P_i-\bar{P}\right)\left(O_i-\bar{O}\right)}{\sqrt{\sum_{i=1}^{N}\left(P_i-\bar{P}\right)^2 \sum_{i=1}^{N}\left(O_i-\bar{O}\right)^2}} \tag{5}$$

$$E_f = \left[\sum_{i=1}^{n}(O_i - \bar{O})^2 \ - \ \sum_{i=1}^{n}(P_i - O_i)^2\right] \ / \ \left[\sum_{i=1}^{n}(O_i - \bar{O})^2\right] \tag{6}$$

Where O_i is the OBC of asphaltic concrete mixtures observations, P_i is the predicted value, N is the total number of data points in validation, $\bar{O}$ is the mean value of observations, and $\bar{P}$ is the mean value of predictions.

3.1 NN models

ANNs are data intensive. ANNs learn the underlying physics of the system of interest from the training samples, which are basically the cause-effects samples. Therefore, the number of training samples significantly influences the network's predictive performance [13]. Increasing the number of training samples provides more information about the shape of the solution surface(s) and thus increases the potential level of accuracy that can be achieved by the network. Having too few data samples will lead to poor generalization by the network. An optimal data set for training would be the one that fully represents the modeling domain and has the minimum number of repetitive samples (i.e., identical inputs with different outputs) in training. Since nearly 70% of the whole data were randomly chosen for model calibration (training) and rest 30% were kept for model validation.

3.1.1 BPNN models

The determination of the optimal network architecture for a given task remains an open research question.

Network size is determined by input number *m*, the number of hidden layers and the number of neurons in the hidden layers. If different network sizes have similar values for the error function, the smallest network size is optimum. Many techniques are proposed to determine optimum neural networks such as the ad hoc approach, the dynamic approach, and the distribution approach. In this study trails were used to characterize the deterministic and random properties of the data to determine optimum neural networks. The input data for neural networks model has been chosen as percentage of course aggregate (igneous rock), percentage of fine aggregate (natural sand), and percentage of filler and the output is the percentage of OBC. The data were collected from the city of Makkah in Saudi Arabia and the properties of these materials and gradations, job mix formula, mixture design was based on the specifications of Ministry of transportation in Saudi Arabia (MOT), so other factors were assumed to be constant.To finally determine the optimum size of neural networks based on the calculated input numbers, the networks with one hidden layer were used for training and test by changing neuron size in the hidden layer. Model results for different BPNN architectures are presented in Table 3.

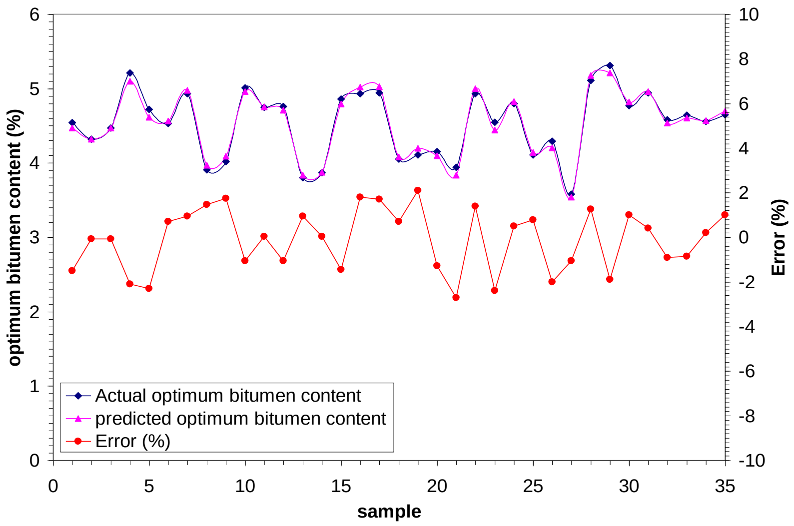
Table 3: Evaluating performance of models.

Scheme	Topology	Training Epochs	E_f (%)	RMSE	R
FF-BP	3-2-1	540	84.854	0.227	0.941
	3-3-1	1920	91.523	0.163	0.960
	3-4-1	3680	93.046	0.147	0.964
	3-5-1	3910	93.241	0.144	0.965
	3-6-1	640	87.169	0.216	0.943
	3-7-1	1960	92.048	0.158	0.961
	3-8-1	**5750**	**94.363**	**0.082**	**0.979**
	3-9-1	2590	90.957	0.169	0.956
	3-10-1	3900	92.747	0.150	0.965
	3-8-1-1	1680	89.938	0.180	0.966
	3-8-2-1	1420	92.428	0.153	0.960
	3-8-3-1	2820	92.830	0.118	0.963
	3-8-4-1	**15200**	**96.370**	**0.052**	**0.989**
	3-8-5-1	3590	93.324	0.144	0.964
	3-8-6-1	7760	95.217	0.109	0.970
	3-8-7-1	1490	93.581	0.141	0.966
RBF	RBF (0.35)	110	87.869	0.155	0.962

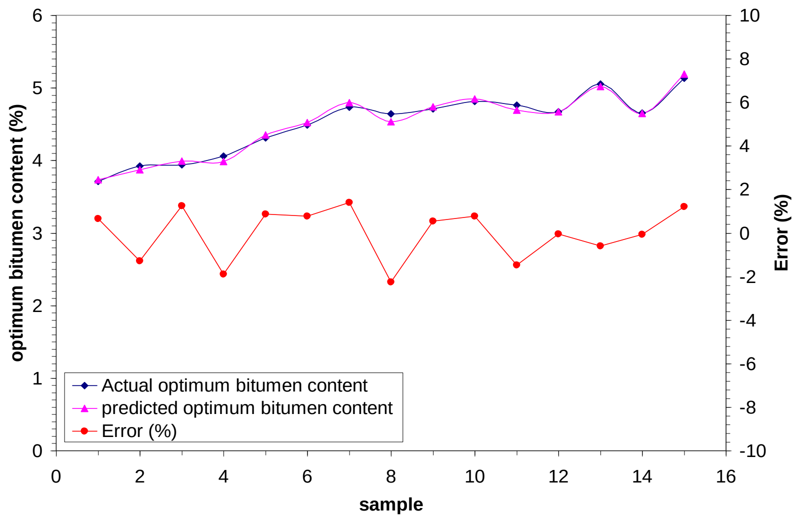
The bold lettered rows show superior results among other ANN structure of same category. The 3-8-1 refers to three neurons in input layer, six neurons in hidden layer, and one neuron in output layer. The 3-8-4-1 refers to three neurons in input layer, six neurons in first hidden layer, five neurons in second hidden layer, and one neuron in output layer. RBF (0.35) refer to RBF network with spread 0.35.

The optimum bitumen content prediction efficiencies of almost all cases of a three-layer BPNN having different neurons in the hidden layer were found more than 85%. In order to explore a BPNN having optimum generalization ability, the BPNN model with different architecture (two hidden layer) was used. Overall, it could easily be observed from Table 3 that BPNN having architecture 3-8-1 (three neurons in input and eight neurons in first hidden layers, and one neuron in output layer) for one hidden layer case produced the best result in this study. BPNN having a structure 3-8-4-1 (three neurons in input and eight neurons in first hidden layers, four neurons in second hidden layer and one neuron in output layer) produced optimum result for the available inputs.

In order to determine the optimum configuration of the four-layer BPNN (3-8-4-1) model, a sensitivity analysis was performed by varying the network parameters, learning rate that minimize the error in test scenarios. The applied network parameters for the learning rate, the momentum and the input noise were found 0.6, 0.9 and 0.03, respectively. The best training and testing tolerance were 0.01 and 0.015, respectively. For the developed BPNN (3-8-4-1) model, Fig. 4 shows the comparison of predicted and experimental values of optimum bitumen content for trained and tested instances. Figure 5 represents the scatter diagram of predicted and experimental values of optimum bitumen content for trained and tested instances. The prediction can be seen as fairly close to the corresponding actual values of optimum bitumen content. For trained data, It can be observed that a maximum absolute error of 2.4%, a minimum absolute error of 0.071% and the average absolute error of 1.185% were obtained for OBC prediction. Also For tested data, It can be observed that a maximum absolute error of 1.9%, a minimum absolute error of 0.04% and the average absolute error of 1.01% were obtained for OBC prediction. The correlation coefficients of 0.989 and 0.992 were obtained for the testing data of optimum bitumen content prediction.



a. Trained data

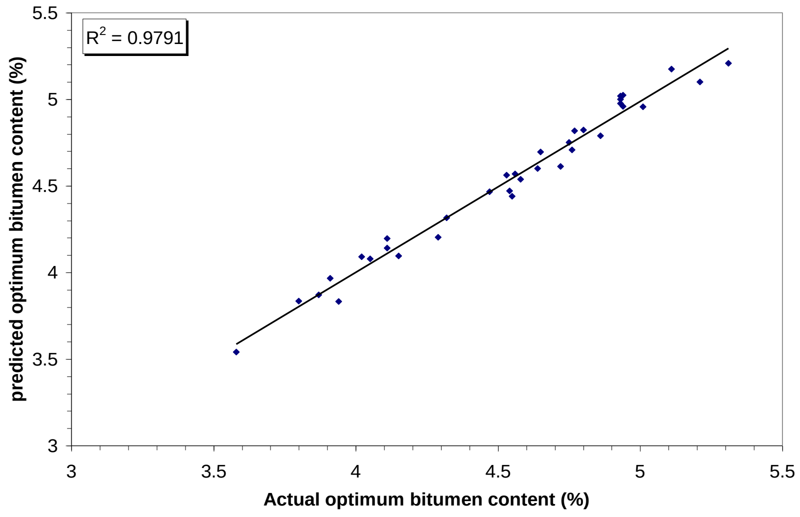


b. Tested data

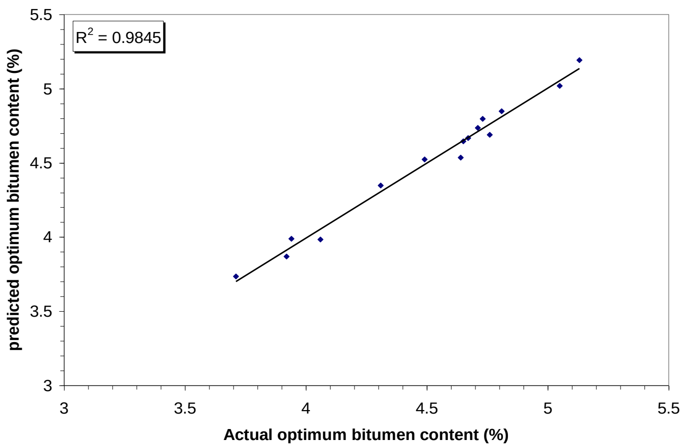
Figure 4: Comparison of predicted and actual values of OBC of asphaltic concrete mixtures for BPNN (3-8-4-1) model.

3.1.2 RBF models

RBF neural network was employed but the prediction ability of the RBF neural network was found poor in terms of the four ANN performance efficiency terms. The probable reason might be the three-layer structure of the neural network, which is not capable of generalizing the process from the available data. The *R* -value during training phase was found to be greater than 0.95. This infers that RBF network, which, in general, requires many neurons in the hidden layer for high dimensional input spaces [12], trained well; however, it lost the generalization ability. Moreover, the neurons in the hidden layer of RBF network have localized receptive fields because they respond to inputs that are close to their centers. This is in contrast to BPNN, where the sigmoid function of hidden layer neuron creates global response [12]. This could be another reason why the RBF network lost its generalization ability in this problem. Sensitivity analysis was performed to find a good spread. The RBF was tested for different spreads from 0.1 to 1.0 and the range varied with an increment of 0.05. This means that 19 models were evaluated to find one best model for predicting OBC of asphaltic concrete mixtures. Among the resulting models, only one RBF network with the smallest RMSE was selected as the best model. The best RMSE initially decreases and then increases with increasing spread. One optimal model achieved at spread equal 0.35 has the RMSE of about 0.155 for training phase (see Table 3).



a. Trained data



b. Tested data

Figure 5: Scatter of predicted and experimental values of OBC of asphaltic concrete mixtures for BPNN (3-8-4-1) model.

3.2 ANFIS model

The other prediction model developed is ANFIS model. For ANFIS simulation, the data sets were divided into three groups. The first one was used as training data, the second one as checking data, and the third one as testing data. The training and checking data were used for estimating membership function parameters and controlling the possibility of falling model into the over fitting problem, respectively. After developing those ANFIS models, the testing data was used for validating the developed ANFIS models. Since it is important that the ANFIS is kept as fast and efficient as possible, a subtractive clustering method (Chiu, 1994) was used to estimate the number of clusters and cluster centers in the data set including percentage coarse aggregate, percentage of fine aggregate, percentage of filler and the percentage of OBC. This helps find an initial FIS in which the number of fuzzy rules is manageable. Subsequently ANFIS was used to tune this initial FIS. The ANFIS model was used to predict the OBC) of asphaltic concrete mixtures. These predicted values were compared with the experimental data to see how well the ANFIS model performs. Figure 6 shows the scatter diagram of predicted and experimental values of optimum bitumen content for trained, checked, and tested instances. As can be seen from figure, ANFIS has performed quite well in predicting the OBC asphaltic concrete mixtures. For trained data, it can be observed that a maximum absolute error of 2.97%, a minimum absolute error of 0.098% and the average absolute error of 1.56% were obtained for OBC prediction. Also, for tested data, it can be observed that a maximum absolute error of 1.95%, a minimum absolute error of 0.06% and the average absolute error of 1.13% were obtained for O BC prediction. The correlation coefficients of 0.976 and 0.99 were obtained for the testing data of optimum bitumen content prediction. Results of the testing of ANFIS were indicated almost same performance of the BPNN (3-8-4-1) model.

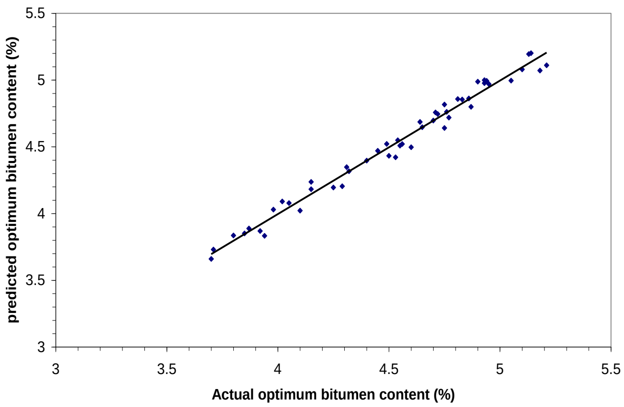


Figure 6: Scatter of predicted and experimental values of OBC of asphaltic concrete mixtures for ANFIS model.

4. Conclusions

The following major conclusions can be drawn from this investigation:

i. With the available information, the four layers BP network with a 2:1 neuron ratio between the first hidden layer and the second hidden layer produced better prediction performance efficiencies with an accuracy of 96.37%. The prediction model was fairly close to the corresponding actual values of optimum bitumen content with the average error of 1.1854% and 1.01% for trained and tested data respectively. The correlation coefficients of 0.989 and 0.992 were obtained for the training and testing data for prediction of the BOC of asphaltic concrete mixtures.

ii. Transportation and highway engineers may use the four-layer BPNN (3-8-4-1) model to predict the optimum bitumen content of asphaltic concrete mixtures without conducting costly and time-consuming experimental tests.

iii. Among the BP and RBF networks employed for this problem, Both the BP models were found to be superior to the RBF network and the four-layer BPNN (3-8-4-1) was found to be superior to the others in prediction of the bitumen content of asphaltic concrete mixtures For BP type of network, the network structure was found to be more important, while for RBF type network, the spread was more effective.

iv. It was found that the ANFIS has performed quite well in predicting the OBC of asphaltic concrete mixtures. Also, Results of the testing of ANFIS were indicated almost same performance of the BPNN (3-8-4-1) model.

Acknowledgment

The author would like to express his high appreciation to the referees of the paper for their critical review as well as their valuable comments that improved the paper to its present form.

References

[1]. J.F. Dias, L. Picado-Santos, and S. Capitão. "Mechanical performance of dry process fine crumb rubber asphalt mixtures placed on the Portuguese road network." Construction and Building Materials, vol. 73, pp. 247-254, 2014.

[2]. A. Pasandín, and I. Pérez. "Overview of bituminous mixtures made with recycled concrete aggregates." Construction and Building Materials, vol.74, pp. 151-161, 2015

[3]. L.Wang, , et al., "Advances in Pavement materials, design, characterisation, and simulation." Road Materials and Pavement Design, vol. 18(3): pp. 1-11, 2017.

[4]. N. Baldo, E. Manthos, and M. Pasetto, "Analysis of the Mechanical Behaviour of Asphalt Concretes Using Artificial Neural Networks." Advances in Civil Engineering, 2018

[5]. M. Alawi, and M. Rajab. "Determination of optimum bitumen content and Marshall stability using neural networks for asphaltic concrete mixtures." in Proceedings of the 9th WSEAS International Conference on Computers, World Scientific and Engineering Academy and Society (WSEAS), 2005.

[6]. J.-S. Jang, "ANFIS: adaptive-network-based fuzzy inference system." IEEE transactions on systems, man, and cybernetics, vol. 23(3), pp. 665-685, 1993

[7]. M.G. Schaap, and F.J. Leij, "Using neural networks to predict soil water retention and soil hydraulic conductivity." Soil and Tillage Research, vol. 47(1-2), pp. 37-42, 1998.

[8]. S. Haykin, "Neural Networks, Comprehensive Foundation," Piscataway. NJ: IEEE Press, 1999.

[9]. H.R. Maier, and G.C. Dandy, "The effect of internal parameters and geometry on the performance of back-propagation neural networks: an empirical study." Environmental Modelling & Software, vol. 13(2), pp. 193-209, 1998

[10]. W. Dias, and S. Pooliyadda, "Neural networks for predicting properties of concretes with admixtures." Construction and Building Materials, vol. 15(7): pp. 371-379, 2001.

[11]. J. Haddadnia, K. Faez, and M. Ahmadi, "A fuzzy hybrid learning algorithm for radial basis function neural network with application in human face recognition." Pattern Recognition, vol. 36(5): pp. 1187-1202, 2003.

[12]. M.T. Hagan, H.B. Demuth, and M.H. Beale, "Neural network design," PWS Pub. Co., Boston, 1996.

[13]. I. Flood, and N. Kartam, "Neural networks in civil engineering. I: Principles and understanding." Journal of computing in civil engineering, vol. 8(2): pp. 131-148, 1994.