Pakistan Journal of Information and Technology 1(2): 180-187, 2002

Modeling Deflection Basin Using Neurofuzzy in Backcalculating Flexible Pavement Layer Moduli

Mehmet Saltan
Civil Engineering Department, S. Demirel University
32260 Isparta/Turkey

Abstract: Flexible pavement surface condition data play a principle role in the evaluation of pavement structural capacity and in the design of pavement rehabilitation programs. Flexible pavements are affected by moving vehicles, climate and other environmental factors. As a result of these factors, the pavement starts to deteriorate. In order to prevent further deterioration, a maintenance program should be carried out at right time and right places. For the determination of the structural carrying capacity of the pavement, non-destructive testing equipments are used. In such a process, the most important thing is to analyse the collected data. A backcalculation procedure is carried out for back-calculating elastic modulus for each layer effective in the pavement life. The input data are usually restricted to the pavement surface deflection or its basin obtained by Nondestructive testing as the Falling Weight Deflectometer (FWD). Generally, linear elastic and finite element based programs are used for backcalculation, but they are both time consuming. It is also important to simulate deflection basin realistically in backcalculating pavement layer moduli. For this purpose, NeuroFuzzy method is used for simulating deflection basin during the course of this study. Results indicate that the NeuroFuzzy can be used for backcalculation of flexible pavement layer moduli with great improvement and accuracy.

Key Words: NeuroFuzzy, Flexible Pavements, Backcalculation, Modulus, Deflection Basin

Introduction

Flexible pavement surface deflection data play a principle role in the evaluation of pavement structural strength and in the design of pavement rehabilitation programs. Backcalculation of pavement layer moduli based on surface-deflection measurements has now been widely used to evaluate structural capacity and layer properties of pavements (Himeno et.al. 1990). Pavement surface deflection data under load are essential to diagnose the pavement condition. There are large number of backcalculation procedures and computer programs to evaluate obtained deflection values.

Highway pavements are generally constructed in the form of flexible pavements in which an asphaltic concrete wearing course on the top, base and sub-base layers underneath the wearing course. The base material may be the bituminous mix or granular material that depends on the passage number of heavy vehicles from the considered section of the road. However, the sub-base layer is generally built with the granular material obtained from local quarries. Repeated application of vehicle loads, weather conditions and other factors decrease the serviceability of the pavement. In other words, the comfort decreases while the user costs and the operation cost increases. For this reason, a maintenance program should be set up to decide when and where to make maintenance works. It is important that the maintenance activities should be done at the right time and right places (Saltan *et al* 2000).

Perhaps the most difficult one is to determine the remaining life of the pavement. Many distresses can be seen by eye. In order to determine the remaining life, the pavement should be analysed structurally with material properties for each layer as elastic modulus, Poisson's ratio and thickness of layers. In order to determine the thickness geophysical methods or drilling can be used. However, in order to determine the structural capacity of the pavement, generally, non-destructive test methods (NDT) are used. These are mainly Benkelman beam, road rater, dynaflect and falling weight deflectometer (FWD). Since, the FWD simulates the wheel loading and its dynamic feature, many countries use the FWD (Thompson 1992).

Deflections obtained from the FWD are used to backcalculate the layer material properties which are elastic modulus and Poisson's ratio. In order to determine the material properties, a lot of methods are used. Some of these are linear elastic theory, Finite Element Method (FEM), Artificial Neural Network (ANN). However, NeuroFuzzy can also be used in backcalculating material properties. One of the main objectives of this study is to develop a method which uses NeuroFuzzy to estimate elastic moduli of each layer analytically through the surface deflection measured by the FWD.

Fuzzy Logic: The first publication on fuzzy logic, which also coined its name, dates back to 1965. It was published in the U.S. by Lotfi Zadeh, Professor of Systems Theory at the University of California, Berkeley (Altrock 1995).

The degree of membership can be represented by a continuous function. Fuzzy sets are a true generalisation of conventional sets. The cases $\mu=0$ and $\mu=1$ of the conventional indicator function are special cases of the fuzzy set. The use of fuzzy sets defined by "fuzzy logic". Here, the degree of membership in a set becomes the degree of truth of a statement. The primary building block any fuzzy logic system is the so-called linguistic variable. In the past 30 years, a large number of methods using fuzzy sets have been developed. The first step in a fuzzy logic system design is the definition of the system structure. Inputs and outputs of the fuzzy logic system are defined. The actual control strategy of a fuzzy logic system lies in the definition of the rules (Altrock 1995).

The possible values of linguistic variable are "linguistic terms", mostly just referred to as "terms". These terms are linguistic interpretations of technical Figures. The degree to which the value of a technical Fig. satisfies the linguistic concept of the term of a linguistic variable is called degree of membership. For a continuous variable, this degree is expressed by a mathematical function called membership function (MBF). The membership functions map each value of the technical Fig. to the membership degree in the linguistic terms. Usually, membership functions can be drawn as shown Fig. 1. Many different shapes of membership functions are proposed in scientific literature. However, most practical implementations only use so-called "Standard Membership Functions". Fig. 2 sketches these functions types. Standard membership functions are also normalised; that is, their maximum is always $\mu=1$, their minimum $\mu=0$ (Altrock 1995).

The rules of a fuzzy logic system represent the knowledge of the system. They use linguistic variables as the vocabulary to express the control strategy of a fuzzy logic controller. The result of the fuzzy logic inference is the value of a linguistic variable. The objective a defuzzification method is to derive the non-fuzzy value that best represents the fuzzy value of the linguistic output variable. To obtain the best compromise value for the result of then fuzzy logic inference as real number, the inference results are considered "weights" at the positions of the most typical values of the terms. The best compromise is where the defuzzified value balances the weights. In a fuzzy logic system, the individual rules represent local behaviour. That is, each fuzzy describes the reaction to a certain situation (Altrock 1995).

Fuzzy Rule Base: For control purposes, fuzzy sets can be used to set up rules of the following forms:

R: IF the value of variable X_1 is "large" and variable X_2 is "medium" **THEN** the result Y is "small" (1)

This statement resembles more closely human thinking than any explicit mathematical rules. Therefore, fuzzy control rules can be used for modelling the behaviour of a human expert. Besides it is also very effective in relating a set of outputs to a set of inputs without specifying a mathematical model. Instead a "fuzzy inference procedure" becomes dominant. In the modelling of human expert thinking first the input variables are specified by fuzzy subsets such as "large" and then fuzzy rules similar to equation (1) are developed on the basis of the experts' knowledge and experience.

In the fuzzy inference method sets of corresponding input and output measurements are provided to the membership functions in logical expressions is called fuzzy system and it learns how to transform a set of inputs to the corresponding set of outputs through a **F**uzzy **A**ssociative **M**ap (**FAM**) which is also called as the **F**uzzy **A**ssociative **M**emory (Kosko, 1992). Fuzzy logic does not provide a rigorous way for developing or combining fuzzy rules which can be achieved through many ways. The one of the ways is outlined below:

First the input and output variables are divided into a number of subsets with simple triangular fuzzy membership functions. Generally, there are n^m fuzzy rules where n and m are the numbers of subsets and input variables, respectively. The more subsets the greater the accuracy possible but the larger the rule base which is more difficult to construct. For example, in the case, say, of two inputs X_1 and X_2 with m subsets each, the rule base takes the form of an output Y_k ($k = 1,2,..,m^2$). If there are two input variables as X_1 with "very small" and "small" fuzzy subsets and X_2 say, "medium" and "large" subsets then consequently there will be four rules as

R₁: IF X_1 is very small and X_2 is medium **THEN** Y_1
R₂: IF X_1 is very small and X_2 is large **THEN** Y_2
R₃: IF X_1 is small and X_2 is medium **THEN** Y_3
R₄: IF X_1 is small and X_2 is large **THEN** Y_4

For each triggered rule the membership degree for inputs are computed and these are multiplied to give the weight W_k to be assigned to the corresponding output Y_k. Hence, the weighted average of the outputs from four rules gives a single output, y, as

$$y=\frac{\sum_{k=1}^{4} W_k Y_k}{\sum_{k=1}^{4} W_k} \qquad (2)$$

Thus once the rule base is set up, values of the output can be computed from equation (2) for any combination of input variables fuzzy subsets. A very common method in deciding about the fuzzy rule base is to use sample data and derive the necessary rule base by the fuzzy inference procedure. This involves computing the weight of each rule triggered, accumulating weights and outputs for each rule and finally computing the weighted output for each rule. The following remarks are very helpful in any fuzzy logic (Kiszka, *et al* 1985).

The fuzzy system works best when the rules linking outputs to inputs can be accurately specified.

Sets of rules can be obtained from operating data by the fuzzy inference procedure but these are not quite as good as those derived from accurate results. However, they can be improved by giving greater weight to inputs with larger membership functions and incorporating any insight from relevant theory and especially from experience.

The system is robust in that some rules can be left out or can contain errors without seriously compromising performance.

A fuzzy rule base can be achieved step-by-step from sets of input and output data as follows :

a try to model the problem with minimum number of input variables.

b divide the range of each input variable into a number (usually 4-8 in practice but into m in general) parts to give fuzzy subsets each with a triangular membership function. Theoretically, the optimum number of fuzzy subsets can be found by minimising the total squared error between the observations and predictions. However, similar to the number of subclasses for histogram construction in statistics where rather subjectively depending on the expert view, the number is chosen between 5-15, the number of fuzzy subsets has established empirically between 4 and 8 in practical studies.

c For each data point m (one value for each X_1, X_2 and Y) compute the membership values for X_1 (W_i) and X_2 (W_j) in each of the fuzzy subsets. Only one or two of each will be nonzero. Set membership values of less than 0.5 to zero.

d Compute the weight of each rule for data point m by multiplying the membership values of X_1 and X_2 that correspond to that rule and squaring the result

$$w_k = \left[(w_i)(w_j)\right]^2$$

e Store the output Y_k along with the complete set of rule weights W_k.

f Repeat with all the other data points.

G Compute the weighted average with expression similar to equation (2), (Kiszka, *et al* 1985).

To enhance fuzzy logic systems with learning capabilities we can integrate neural network technologies. The combination of fuzzy logic and neural network technology is called "NeuroFuzzy" and combines the advantages of the two technologies (Altrock 1995).

Artificial Neural Networks: Artificial neural networks (ANN) are widely used in a variety of practical tasks from process monitoring, fall diagnosis and adaptive human interference to natural events and artificial intelligence such as computers (Dimitrova, 1996). They are very important in control system applications because of their universal mapping characteristics and learning ability. ANN process can be considered as a black-box modelling with a set of input factors and output variables which are a result of input factors treatment through a systematic neural network. The first appearance of ANN concept in the literature is due to McCullough and Pits (1943) who suggested the cell model. In such a model, ANNs are exemplified as a set of logical statements. Later, many researches concentrated their attention on the learning ability of human and its modelling (Hebb, 1949) which can be accounted as the pioneering work on ANNs. However, actual leaps in the ANN development appeared towards 1980 through various researches (Hopfield, 1982).

Neural network uses a number of simple computational units called "neurons", of which each tries to imitate the behaviour of a single human brain cell. Each neuron in a neural network processes the incoming inputs to an output. The output is linked to other neurons. Some of the neurons form the interface of the neural network. The neural network shown in Fig. 3 has a layer for the input signals and one for the output signals. The information enters the neural network at the input layer. All layers of the neural network process these signals through the network until they reach the output layer. The objective of a neural network is to process the information in a way that is previously trained. Training uses either sample data sets of inputs and corresponding outputs or a teacher who rates the performance of the neural network. For this training, neural networks use so-called learning algorithms. Upon creation, a neural network is dumb and does not exhibit any behaviour at all. The learning algorithm then modifies the individual neurons of the network and weight of their connections in such a way that the behaviour of the network reflects the desired one (Altrock 1995).

Initially, ANN can be divided into two parts as architecture and neurodynamics (functional properties). The former defines the structure of the network as the number of artificial neurons and their interconnectivity whereas the latter includes their properties as to how the neural network learns, recalls, associates and continuously compares new information with existing knowledge, and how it classifies new information and development of new classifications, if necessary. ANN architecture includes many interconnected neurons or processing elements, with familiar characteristics such as inputs, synaptic strengths, activation, output and bias (Sönmez and Şen, 1998).

In general, a neuron has n inputs as x_j, ($j=1,2,\ldots,n$) which show the source of input signal. Each input is weighed before reaching the main body of processing element (artificial neuron) by the connecting strength or the weight factors, w_j. Hence, the signal transferred through the connection strength is equal to a portion of the original signal as $w_j\, x_j$. On the other hand, for the neuron to produce a signal, the input signal to a neuron must exceed a threshold value, *T*, and in addition to this it has, in general, a bias term *B*. After the effects of the bias and the threshold on the weighted signal a nonlinearity function, *F*, i.e. activation *R*, enters this nonlinear unit and then applications use sample data training. After completion of learning, the neural network is ready to use. This is called the working phase. As a result of training, the neural network will output values similar to those in the sample data sets when the input values match one of the training samples. For input values in between, it comes out as completely treated output, *O*. Of course, this output may be an input for some other neurons. If there are many neurons in a network, then each neuron is called as node within the network. If there are m modes in a network then the above referred procedures will work for each one of them. In order to distinguish between each neuron the

Fig. 1. Fuzzy Subset Membership Functions

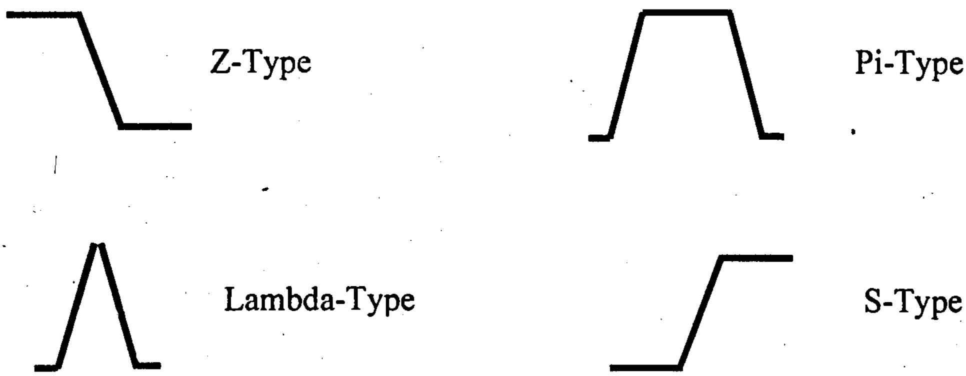

Fig. 2: Standard Membership Functions

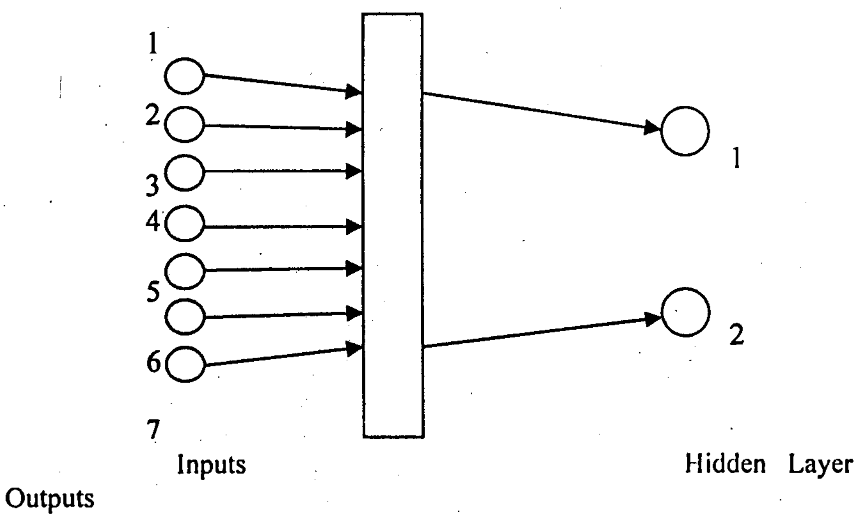

Fig. 3: The ANN Architecture

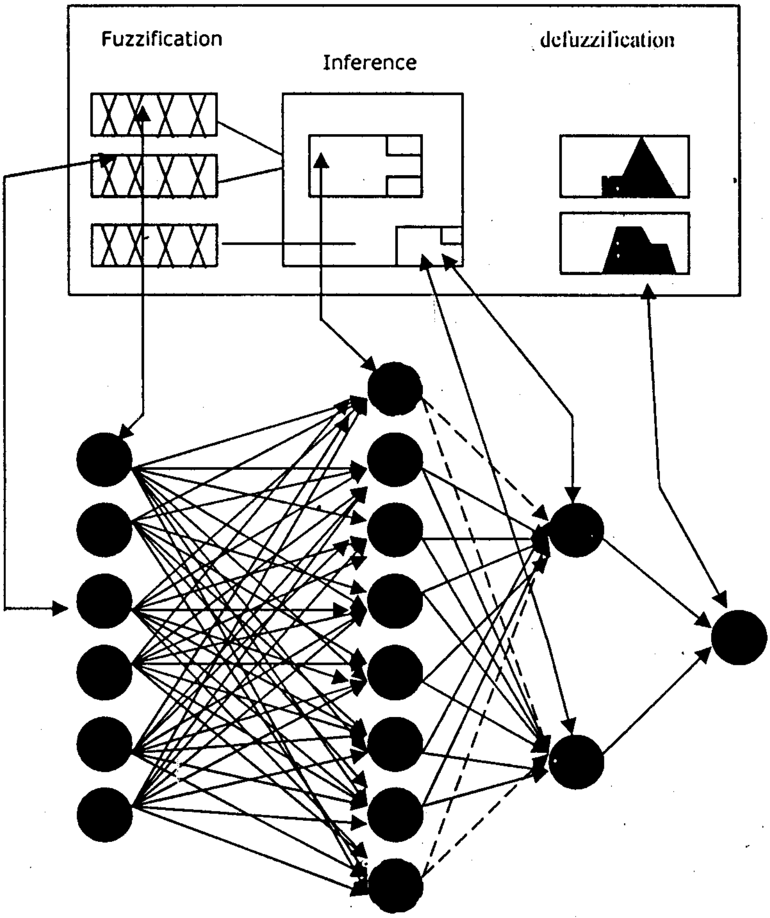

Fig. 4: Neuro Fuzzy Technologies Map

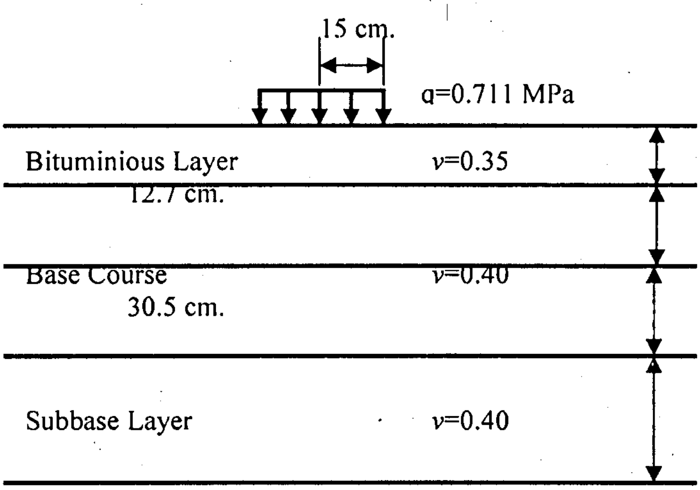

Fig. 5: Example

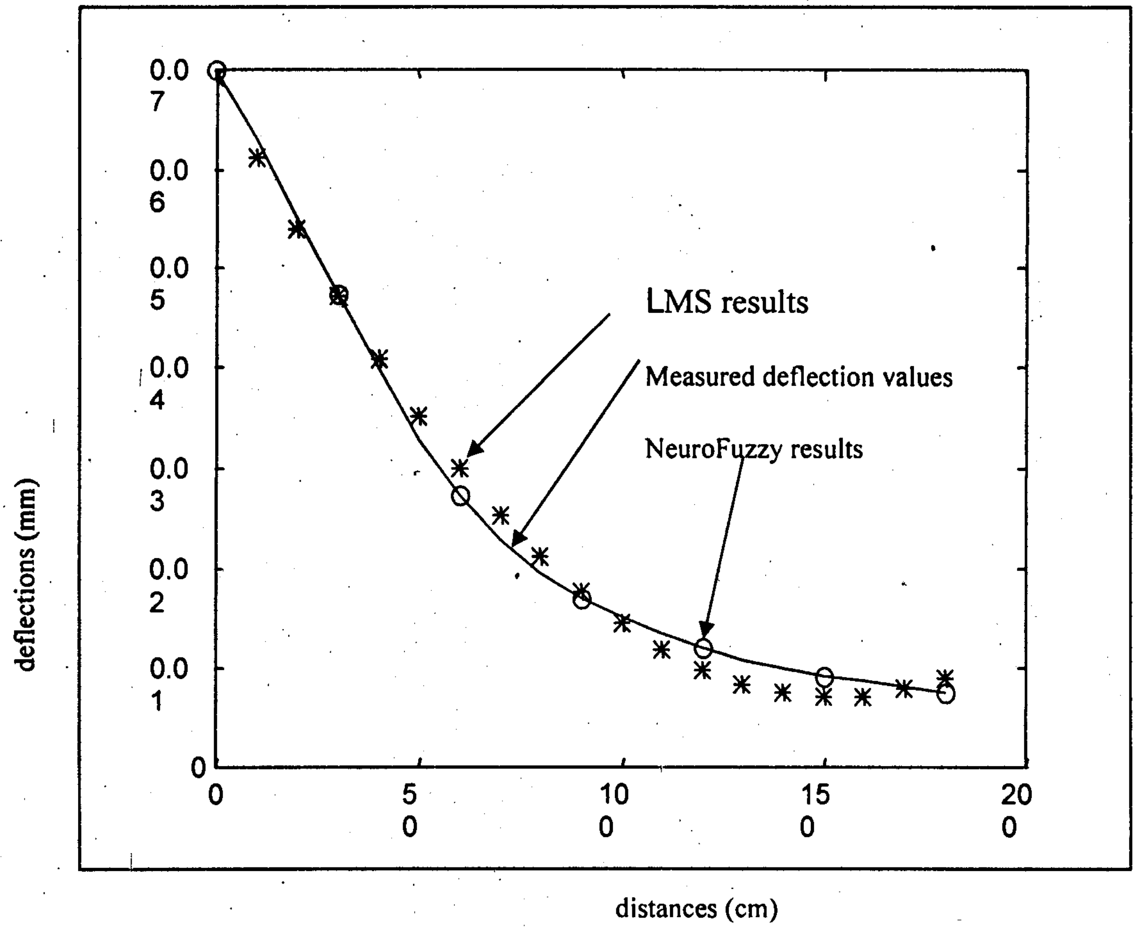

Fig. 6: Comparison of LMS, NeuroFuzzy and Measured Deflections

Table 1: Deflection Values for Example

	d_1	d_2	d_3	d_4	d_5	d_6	d_7
Deflections(cm.)	0.0155	0.011	0.0071	0.0058	0.0050	0.0044	0.0037
Distances(cm.)	30.5	30.5	30.5	30.5	30.5	30.5	

subscript *i* will be used. Accordingly, inputs, weights, activation signals, output, threshold and nonlinear function will all have identification subscript, *i*. The transfer function in an ANN is given by the following relation:

$$O_i = F_i\left(\sum_{j=1}^{n} w_{ij} x_{ij}\right) \tag{3}$$

with the neuron's firing condition as

$$\sum_{i=1}^{n} w_{ij} x_{ij} \geq T_i \tag{4}$$

where the subscripts *i* and *j* represent the neuron in question and the inputs from the neurons.

The reason for including the nonlinearity function is for ensuring the neurons bounded response. This means that the actual response of the neuron is conditioned or damped as a result of large or small activating stimuli and thus is controllable. It is well known that in order to hear a sound as twice as loud, and actual increase in sound amplitude of about 10 times is necessary. This shows almost logarithmic response of the ear.
Two of the most used nonlinearities are the hard limiter and the sigmoid (as expressed in Equation 5) where *x* is the variable and *F*(*x*) is the activation function. Most of the limiters have upper and lower limits as ±1, or 0 and 1. In an actual ANN application, it is up to do user to choose the bound values. However, the sigmoid is very popular because it is bounded,

monotonic, has a simple derivative and nonlinear. The harder limit is not monotonic, and it has discontinuity at the origin. Although it is not easily differentiable, but linear with its upper and lower bounds.

$$F(x) = \frac{1}{1+e^{(-x)}} \tag{5}$$

The first step in designing a neural network solution is teaching the desired behaviour. This called the learning phase. Here, we can use sample data sets. Since neural network is mostly used for complex applications where no good mathematical models exist, and rating the performance of a neural network is hard in most applications, most approximates output values. In the working phase, the behaviour of the neural network is deterministic. That is, for every combination of input values, the output value will always be the same. During the working phase, the neural network does not learn. This is important in most technical applications to ensure that the system never drifts to hazardous behaviour.

Combining Neural Network and Fuzzy Logic: The key benefit of fuzzy logic is that it lets we describe desired system behaviour with simple "if-then" relations. In many applications, this gets you a simpler solution in less design time. In addition, we can use all available engineering know-how to optimise the system performance directly. While this is certainly the beauty of fuzzy logic, at the same time it is a major limitation. In many applications, knowledge that describes desired system behaviour is contained in data sets. Here, the designer has to derive the "if-then" rules from the data sets manually, which requires a major effort with large data sets. When data sets contain knowledge about the system to be designed, a neural network promises a solution because it can train itself from the data sets. However, only a few commercial applications of neural network exist. This is in contrast to fuzzy logic, which is a very common design technique (Altrock 1995).

Both neural network and fuzzy logic are powerful design techniques. Neural network can learn from data sets, while fuzzy logic solutions are easy to verify and optimise. Combining the explicit knowledge representation of fuzzy logic with the learning power of neural network, we can get NeuroFuzzy. The error back propagation algorithm became the standard for most neural network implementation due to its high training performance. First, it selects one of the examples of the training data set. Second, it computes the neural network output values for the current training example's inputs. Third, it compares these output values to the desired output value of the training example. Finally, The difference, called error, determines which neuron in the network shall be modified and how. The mathematical mapping of the error back into the neurons of the network is called error back propagation (Altrock 1995).

To determine which neuron has what influence, the error back propagation algorithm mathematically differentiates the transfer functions of the neurons.

NeuroFuzzy development tools use extended fuzzy logic inference methods. The most common approach is to use so-called fuzzy associate memories (FAMs). A mathematical frame use of a modified error back propagation algorithm with fuzzy logic (Altrock 1995). work exists that maps FAMs to neurons in a neural network (Fig. 4). This enables the

Backcalculation Using Neurofuzzy: Non-destructive testing (NDT) enables the use of a mechanistic approach for pavement design and rehabilitation since in-situ material properties can be backcalculated from the measured field data through appropriate analysis techniques. Backcalculating layer moduli from pavement deflection bowls appears to be a promising method of determining the performance of in-service pavements (Huang 1993, Kang 1998).

The evaluation of material properties of existing in-service pavements is a fundamental problem in pavement engineering. Till now, a lot of highway agency all over the world used the traditional simple methods such as linear elastic theory and equivalent layer thickness. Using this simple methods, it is impossible to evaluate the material properties of in-service pavements in a realistic manner. But finite element method is highly efficient in nonlinear formulations for it can easily accommodate changes in material properties, allowing for variations in both the vertical and horizontal directions. Finite element method employs a process of discretisation whereby the structure is sub-divided into a number of elements, connected at the nodal points. Each element, therefore, has an elastic stiffness consistent with its stress level. Nowadays, finite element method is increasingly used paralleling new technologies.

In this paper, a newly developed backcalculation program (SDUFEM) is described. SDUFEM is very user-friendly program in which backcalculation layer moduli for flexible pavements based on deflections measured by a falling weight deflectometer (FWD). SDUFEM uses the finite element method for calculating pavement deflections under a FWD loading. The problem in using any finite element program is to prepare mesh data. In order to facilitate the mesh preparation, a mesh generator was developed (Saltan 1999).

The deformed shape of a pavement when subjected to a vertically oriented surface loading is known as a deflection basin. The program attempts to match the measured deflection basin for a system with unknown layer moduli with the theoretical deflection basin for a system with known layer moduli. This process is commonly named as backcalculation of layer moduli. A deflection bowl obtained from the deflections using Least Squares Method (LSM) is compared with measured deflection bowl. Both measured and computed deflection bowls are generated using LSM. In general, seven deflections are used to fit a bowl with LSM. In backcalculation process of pavement layer elastic moduli, required deflection value is used from deflection bowl. The elastic modulus which is obtained from backcalculation process will be more realistic when the deflection bowl is modelled as convenience to the real values.

In the program, for modelling both measured and computed deflection basins, NeuroFuzzy method was added. Till now, least squares method is widely used to simulate deflection basin. Generally, many researchers have used the exponential and polynomial formed relationships in modelling deflection basin using LSM.

Initially, system mesh is generated by mesh generator. Then, program reads input data sets which include: system mesh data, FWD load force and load radius, pavement layer thicknesses, Poisson's ratio, seed moduli of layers, density of pavement materials, deflection data, percent tolerance to stop deflection matching process. Deflections for the given FWD load and load radius are calculated by using finite element technique.

In the operation of backcalculation, deflection values of the points that are different from the points of which deflection values are already measured or calculated must be taken into consideration. In this way, the forms of the measured or calculated deflection values must be found, and the deflection values of any points must be calculated according to this form. The calculated deflections are compared to measured deflections. If the calculated deflections don't match to measured deflections, moduli values are adjusted by using correction factors.

For the backcalculation analysis, an initial elastic modulus are set up for each layer. The FWD loading is then applied on the mesh vertically and deflections are compared with error functions below. If the error is not in the acceptable range, the elastic modulus has to be changed until the error function is satisfied. Following the satisfaction, the elastic modulus for each layer is assumed as known quantities. Hence, a forward analysis is carried with the obtained values, in order to find the tensile strain underneath the bituminous mixture and vertical strain on the subgrade. Results obtained from the forward analysis are entered into the fatigue and plastic deformation graphics. From these two graphics, remaining life of the pavement is then determined. Finally, a decision is made whether the overlay is necessary or not. For an objective decision, the following relative error square summation, *RSS*, definition is employed.

$$RSS = \sum_{i=1}^{s}\left(\frac{d_i^m - d_i^h}{d_i^m}\right)^2 = \sum_{i=1}^{s}\left(1-\frac{d_i^h}{d_i^m}\right)^2 \qquad (6)$$

where d_i^h = calculated deflections in i th geophone; d_i^m = measured deflections in i th geophone, and *s*= sensor number from i to *s*.

Most of the backcalculation programs have limitations on the maximum allowable number of layers in their analyses. In addition to mesh generation, SDUFEM have no such limitations.

Example: Fig. 5 is showing a numerical example which the form of deflection basin is determined using NeuroFuzzy in backcalculation process. In this example, a backcalculation study was applied on a pavement with four layers (Fig. 5). Subbase was stabilised with cement. Deflection values measured on this pavement is given in Table 1.

As can be shown in Fig. 6, NeuroFuzzy results are almost same as the measured deflection values. So, we can say that we can model the deflection basin using NeuroFuzzy in realistic manner.

Conclusion

A truck moving over a pavement applies a load pulse that is transmitted through the pavement layers. The pavement layers respond differently to this load pulse, depending on the material types of each layer.

According to load applications, deflection basin is obtained. Layer elastic moduli are then backcalculated using deflection basin and pavement characteristics. The use of backcalculated moduli is essential to the application of mechanistic principles to pavement evaluation. Backcalculation techniques and software have advanced greatly in recent years. In spite of that development, many problems as modelling the deflection basin are still encountered.

In this study, a finite element program including NeuroFuzzy in modelling deflection basin is mentioned. A pavement structure was analysed using this finite element program. NeuroFuzzy results are almost same as the measured deflection values, whereas Last Squares Method (LMS) results differ from measured deflections. So, we can say that we can model the deflection basin using NeuroFuzzy in realistic manner. And NeuroFuzzy is not required long computation time.

References

Altrock, C.V., 1995. Fuzzy Logic and NeuroFuzzy Applications Explained. Prentice Hall, Englewood Cliffs, NJ.

Dimitrova, M., 1996. A Neural Compound Neural Network for User Identification In Adaptive Human Computer Interfaces. J. Complex Control Systems.

Himeno, K., T. Maruyama, N. Abe, M. Hayashi, 1990. The use of FWD deflection data in mechanistic analysis of flexible pavements. Third International Conference on Bearing Capacity of Roads and Airfields, pp. 401-410, Trondheim, Norway.

Hebb, D., 1949. The Organisation of Behaviour. Willey, New York, USA.

Hopfield, J.J., 1982. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proc. Natl. Acad. Sci., 79, pp.2554-2558.

Huang, Y.H., 1993. Pavement analysis and design. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, USA, 804 p.

Kosko, B., 1992. Fuzzy thinking: The New Science of Fuzzy logic. Hyperion, New York, N.Y.

Kang, Y.W., 1998. Multi-frequency backcalculation of pavement-layer moduli. Journal of Transportation Engineering, ASCE, Vol.124.no.1, USA, pp. 73-81.

Kiszka, J.B. Kochanskia, M.E. and Sliwinska, D.S., 1985. The influence of some fuzzy implication operators on the accuracy of fuzzy model. Part I. Fuzzy Sets and Syst. **15**, 111--128.

McCulloch, W.S., W. A. Pitts, 1943. A Logical Calculus of the Ideas Immanent in Nervous Activity. Bull. Math. Biophysics, pp. 115-133.

Saltan, M., Tığdemir, M., Saltan, S., 2000. Structural Analysis of Flexible Pavements Using Artificial Neural Network. J. of Inst. Maths.&Comp.Sciences (Comp.Sc.Ser.)1 : 2000 25-35.

Sönmez, I., Z. Şen, 1998. Artificial Neural Network Approach for Natural Atmospheric Event Dynamics And Application In Meteorology. Proc. of 2nd Int. Symp. on Intelligent Manufacturing Systems, pp.325-331, Sakarya.

Saltan, M. 1999. Analytical Evaluation of Flexible Pavements. PhD Thesis, 202 p., Isparta, Turkey.

Thompson, M.R., 1992. Illi-pave based Conventional Flexible Pavement Design Procedure. 7th International Conference on Asphalt Pavements, 1, pp. 222-230, U.K.