An Application of Artificial Neural Networks (ANNs) To the Backcalculation of Flexible Pavement Moduli

Faten Fakher Abdulnibe 1, kassim A. Jassim2

12 University of Baghdad, College of Science, department of Mathematics, Baghdad-Iraq

Abstract-- The role of a falling weight deflectometer (FWD) test lies in the capacity to measure pavement system responses in relation to transient loads. These loads are applied on surfaces of the pavements. A study by ARA Inc. & ERES Consultants Division (2004) indicated that the backcalculation of pavement layer moduli via FWD has continually gained adoption or wide employment, yet it has not gone without limitations. As concurred by Ceylan, Guclu, Tutumluer and Thompson (2005), FWD results are dependent on pavement response static analyses. Indeed, most of the previous studies highlight that the discrepancies between FWD test dynamic nature and the static assumption yield significant errors in moduli (Chatti, Ji & Harichandran, 2004). It has also been established that a number of dynamic solutions exist in relation to pavement response but computational complexities with which these approaches are associated imply that their application in programs of conventional backcalculation is impractical (Goel & Das, 2008). Indeed, it is the limitation of FWD results and the impractical nature of adopting dynamic solutions that have paved way for the application of artificial neural network technologies while seeking to address the backcalculation dilemma with precision. Artificial neural networks constitute simple processing element collections that are highly interconnected and, upon training, could aid in the approximation of inverse functions (Goktepe, Agar & Lav, 2005). Goktepe, Agar and Lav (2005) observed that the approximation is achieved via repeated shows of forward problem solutions. Gopalakrishnan (2010) avowed that the leading advantage associated with artificial neural networks (ANN) concerns the aspect of speed.

Keywords- backcalculation, deflectometer, FWD

1. Background

Indeed, the function of Falling-Weight Deflectometer (FWD) lies in the assessment of flexible pavements’ structure properties in a non-destructive manner. As the FWD results are evaluated, the in-situ pavement layer moduli’ backcalculation is done in relation to the measured deflections (Gopalakrishnan & Khaitan, 2010). To accomplish the backcalculation process, calculated theoretical deflection basins are compared with measured deflection basins for assumed road structures or specific road structures. Indeed, assumed road structures are applied in situations where the structure of the pavement remains unknown (Gopalakrishnan & Thompson, 2004). According to Kim, Kim and Mun (2010), the calculation of theoretical deflection basins is achieved via linear-elastic, static analyses of multi-layer systems. Despite the availability of backcalculation programs that are conventional, it is important that user intervention is enhanced towards the achievement of representative results regarding the structure of the pavement (Mehta & Roque, 2003). In this paper, the main aim is to examine the use of these artificial neural networks in enhancing the backcalculation of FWD-generated data in relation to flexible pavement moduli.

2. Field Analyses

To provide pavement sustainability, accurate maintenance strategies are deemed important to transportation agencies. With imposed loadings aiding in measuring surface deflections via FWD, the resultant non-destructive pavement evaluation has proved critical in informing these agencies based on the resultant structural assessments. Hence, the criticality of pavement layer backcalculation to determine
properties using programs such as ANN cannot be overemphasized. Figure 2 represents a back-propagation neural network, the principles on which backcalculation via ANN works.

![Figure 1: A back-propagation neural network represented](image)

From the figure, BP and AO indicate error back-propagation and directions of activation while $o1$ to $o2$ and $i1$ to $i4$ depict the problem’s output and input variables. It is also worth noting that the hidden layers’ neurons are depicted by $h_{11}$ to $h_{23}$. In figure 3, ANN’s typical processing unit is represented. In the figure, $X_i$ refers to the input signal while $W_{ij}$ constitutes the connection weights. Indeed, the input signal is meant for prior neurons of $N$ number. Similarly, $y_i$ entails the output signal, $net_i$ the net input signal, and $\theta_i$ the activation threshold (Seo, Kim, Cho & Jeong, 2013).

![Figure 2: Illustration of the processing unit](image)
From the direction of activation propagation, the new neuron is reached by input signals emerging from prior processing units. The unit evaluates these signals in relation to their respective connection weights before multiplying these individual input signals by their corresponding connection weights. In so doing, internal activity of neurons is calculated in relation to the input signals’ weighted summation (Sharma & Das, 2008). To calculate the net input signal, the equation used holds:

$$net_i = \sum_{j=1}^{n} (W_{ij}X_j) - \theta_i$$

In situations where the resultant input signal is more than the threshold limit value, the response of the neuron is expected to indicate $y_i$ based on selected transfer functions $f(x)$. In response to the net internal activity, the output response gives:

$y= f(net_i)$

Regarding the resultant functions, they fall into three categories namely: sigmoidal, threshold, and linear. Indeed, the sigmoidal transfer function exhibits more similarity when compared to real neurons (over linear and threshold transfer). With changes in $y_i$ (the output signal value) for the sigmoidal function perceived to lie between 0 and 1, the resultant equation holds:

$$f(x) = \frac{1}{1+e^{-x}}$$

Indeed the back-propagation learning rule while employing ANN for backcalculation lies in the quest to reduce the error or difference between the calculated and the desired output values; translating into supervised learning. As the learning or training process begins, randomly initialized connection weights are provided before the updating stage based on the level of the error. As each of the steps employed in forward propagation ends, an objective function is applied towards the calculation of the error $E_k$. In this case,

$$E_k = \frac{1}{2} \sum_{i} \left( t_i^k - y_i^k \right)^2$$

Notably, neuron $k$ and $i$ data’s actual output is represented by $t_i^k$. As observed above, the calculated error determines the adjustment done to the connection weights (ARA Inc. & ERES Consultants Division, 2004). Similarly, expressing $\Delta W_{ij}$ (the amount of change between $j$ and $i$ neurons) requires calculations of derivatives of error terms in relation to the resultant connection weights. Calculating this change gives:

$$\Delta W_{ij} = -\eta \frac{\partial E}{\partial W_{ij}} = -\eta \sum_k \frac{\partial E_k}{\partial W_{ij}}$$

The learning coefficient, which is expected to exceed zero, is represented by $\eta$. An application of the chain rule gives

$$\frac{\partial E_k}{\partial W_{ij}}$$

Rewriting in the delta term ($\delta_k^i$) gives:
Given the availability of the estimated and actual output signals, the computation of the delta term in the output layer can be done. Regarding the hidden layer, the output signals expected to be sent are unknown. As such, the role of the delta term $\delta_{mk}$ in this case lies in the calculation of the current delta value. Imperative to highlight is that the latter value employs neurons $m$; with the previous $i$-th layer forming its location (Ceylan, Guclu, Tutumluer & Thompson, 2005). Expressing the resultant general delta rule gives:

$$
\delta_{k}^{l} = \left\{ \begin{array}{ll}
\sum_{m} \delta_{mk}W_{lm}f'(net_{l}^{k}) & \text{for output layers} \\
\delta_{mk} & \text{for hidden layers}
\end{array} \right.
$$

The sigmoidal function’s derivative becomes:

$$
f'(x) = f(x) \{1 - f(x)\}
$$

Upon completing these activation propagation steps, the process of back-propagation commences from output layers to the input layers. This process, as observed by Chatti, Ji and Harichandran (2004), is marked by adjustments of link weights in the respective iterations, which are successive. The eventuality is that the activation direction’s outputs become the back-propagation direction’s inputs (Goel & Das, 2008). For iterations that follow, updating new connection weights of $j$ and $i$ neurons gives:

$$
W_{ij}(t+1) = W_{ij}(t) + \eta \sum \delta_{k}^{j}x_{j}
$$

Indeed, the momentum term is represented by $\alpha$ and considers changes in weight in the preceding iterations meant to prevent the trapping in of local minima by the algorithm and, also, to yield oscillations. The modification of bias values is also done in the same manner as the link weights to give:

$$
th_{i}(t+1) = th_{i}(t) + \eta \sum \delta_{k}^{i}
$$

Indeed, the above steps are expected to be repeated for the individual data in training sets in an iterative manner while seeking to achieve the minimum error between the calculated and the desired output. Given that ANN exhibits the capacity to solve complex problems that are resource-intensive in an accurate and fast manner, the trend accounts for its extensive application in pavement problems (Goktepe, Agar & Lav, 2006). Indeed, the adaptive backcalculation approach had its initial applications spearheaded by Meier and Rix (1994), targeting attributes such as dynamic deflection basins, layer properties, and surface wave inversions. Notably, an achievement of correct backcalculated layer moduli demands accurate deflection measurements. However, Goktepe, Agar and Lav (2005) observed that it is unlikely to be realistic to expect an exact match between the systematic deflections responsible for neural network training and the experimental deflections. Hence, a number of measurement errors, which are equipment-related, arise. On the one hand, random errors constitute the noise or random measurement errors that pose the difficulty of reproduction (Gopalakrishnan, 2010). On the other hand, systematic errors are repeatable and, through proper measurement apparatus calibration, they can be minimized. Universally, the FWD test specifications hold that the repeatability error does not exceed 0.08 mils while the respective geophones’ systematic errors do not exceed 2% (Gopalakrishnan & Khaitan, 2010).

3. Conclusion

Given that ANN fosters real-time backcalculation in relation to simultaneous data acquisition and analysis, the emerging benefit is that its application aids in alleviating both the indirect costs of commuting delays and the direct costs of traffic control. Hence, work crews who use ANN as a layer moduli backcalculation tool are likely to be at less risk as they do not necessarily need to work in the middle of traffic lanes, outside their vehicles. However, the need for crew workers to consider sub-grade
thickness during the backcalculation process cannot be overemphasized. Overall, the paper has established that the evolution of ANN as a tool for backcalculating flexible pavement moduli from FWD data promises outcomes such as speed and efficiency. In future, the use of ANN as a tool for the backcalculation of FWD-generated data is poised to stretch beyond the merits of speed and efficiency to foster cost-effectiveness from the commuting and traffic control-related expenses; yet quality is also predicted to remain uncompromised.

4. References

[1]. Ceylan, H., Guclu, A., Tutumluer, E., & Thompson, M. (2005). Backcalculation of full-depth asphalt pavement layer moduli considering nonlinear stress-dependent subgrade behavior. International Journal of Pavement Engineering, 6(3), 171-82

[2]. Chatti, K., Ji, Y., & Harichandran, R. (2004). Dynamic time domain backcalculation of layer moduli, damping, and thicknesses in flexible pavements (pp. 106-116). Michigan: Transportation Research Record

[3]. Goel, A., & Das, A. (2008). Non-destructive testing of asphalt pavements for structural condition evaluation: a state of the art. Journal of Non Destructive Testing and Evaluation, 23(2), 121-40

[4]. Goktepe, A., Agar, E., & Lav, A. (2006). Role of learning algorithm in neural network-based backcalculation of flexible pavements. Journal of Computing in Civil Engineering, 20(5), 370-373

[5]. Goktepe, A., Agar, E., & Lav, H. (2005). Advances in back-calculating the mechanical properties of flexible pavements. Advances in engineering software, 37, 421-431

[6]. Gopalakrishnan, K. (2010). Neural network-swarm intelligence hybrid nonlinear optimization algorithm for pavement moduli back-calculation. Journal of Transportation Engineering - ASCE, 136(6), 528-536

[7]. Gopalakrishnan, K., & Khaitan, K. (2010). Finite element based adaptative neuro-fuzzy inference technique for parameter identification of multi-layered transportation structures. Transport, 25(1), 58-65

[8]. Gopalakrishnan, K., & Thompson, M. (2004). Backcalculation of airport flexible pavement non-linear moduli using artificial neural networks. In Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference 2, 652-57

[9]. Kim, D., Kim, J., & Mun, S. (2010). Normalisation methods on neural networks for predicting pavement layer moduli. Road & Transport Research, 19(3), 38-46

[10]. Mehta, Y., & Roque, R. (2003). Evaluation of FWD data for determination of layer moduli of pavements. Journal of Materials in Civil Engineering (ASCE), 25-31

[11]. Seo, J., Kim, Y., Cho, J., & Jeong, S. (2013). Estimation of in situ dynamic modulus by using MEPDG dynamic modulus and FWD data at different temperatures. International Journal of Pavement Engineering, 14(4), 343-353