USING A BACK-PROPAGATION ALGORITHM TO CREATE A NEURAL NETWORK FOR INTERPRETING ULTRASONIC READINGS OF CONCRETE

A. Lorenzi[1], L. C. P. Silva Filho[1], and J. L. Campagnolo[1]
[1] Universidade Federal do Rio Grande do Sul, Laboratório de Ensaios e Modelos Estruturais (LEME), Porto Alegre, RS, Brasil

Abstract: As widely known, concrete is an essential material in civil engineering. However, its properties can vary considerably, depending on the nature and proportions of its constituents, the construction methods applied to create it, and the loading and environmental conditions to which it will be subjected over time. Therefore, the development of control methods to determine the condition and ascertain the quality of concrete is critical. Ultrasonic methods can play an important role in this area, since they allow us to monitor the density and homogeneity of concrete, providing information about the strength development and the existence of internal flaws and defects. To ensure that the information provided is reliable, expert knowledge is needed for interpreting the ultrasonic data. Computational tools that support the interpretation of ultrasonic (and other NDT) data might facilitate the job, reducing bias and helping specialists to analyze, in a consistent way, the great amount of data generated by the test. In this context, the use of Artificial Neural Networks (ANN) techniques is seen as a viable and adequate strategy to develop such tools. This study is focused on the evaluation of the feasibility of developing a specialist ANN tool. This kind of technique allows the emulation of the though processes of specialists when dealing with uncertainty. Using a neural model, it is possible to establish a non-linear correlation between known input data, such as age and ultrasonic readings and a certain output (in this case, compressive strength, because this is the most used parameter to determine concrete quality). The net was trained using a back propagation algorithm to minimize the mean squared error. The results obtained indicate that, with four layers of perceptrons, the estimation power of the neural network is better than using traditional modeling techniques, such as regression analysis.

Introduction: There are several techniques for modeling the process of converting data into information that try to emulate the human ability to reason. The use of Artificial Neural Networks (ANN) is a new alternative, capable of solving complex problems using an "artificial reasoning system" constructed with basis on the human brain. These computational tools were inspired by the analysis of the neural structure of intelligent organisms and use knowledge acquired through the analysis of previous experiences to develop correlations between known initial conditions and results.

The basic idea is to reproduce the vast array of relationships that are established between individual brain neurons, using different synaptic pathways to determine the output to a certain stimulus. The neurons are the basic building blocks of the human brain, one of the most efficient "processing machines" known to date. Nonetheless, the chemically-based biological neurons are much slower than silicon logic gates. The brain makes up for the slow rate of operation because [3]:

- There are a huge number of nerve cells and interconnections between them inside the brain. Besides, the functions performed by a biological neuron seem to be much broader than those of a logic gate.
- The brain is very energy efficient. It consumes only about 10^{-16} joules per operation per second, comparing with 10^{-6} joules per operation per second for a digital computer.

The brain are a highly complex, non-linear, parallel information processing system, which performs tasks like pattern recognition and visual processing many times faster than the fastest digital computers. However, the human brain might be influenced by emotional or technical bias, and it is not very efficient when dealing with problems of ranking involving a large number of items. Furthermore, it takes a long time to train a human brain to perform to the best of its abilities. There are situations when using an artificial, less efficient, but more organized tool might be useful. In this way, ANNs might alleviate experts and contribute to produce quick and unbiased assessments that might help decision-makers to deal with uncertain and complex matters.

Using Artificial Neural Networks: Due to their nature, ANNs are very useful for analyzing complex problems where the relationships between input and output data are not very well known, such as pattern and speech recognition, machine vision, robotics, signal processing and optimization. They are also useful in fields where there is a high degree of uncertainty, such as market analysis, analysis and control of industrial processes and medical diagnosis. In the case of civil engineering, the ANNs have already begun to be used in problems of structural diagnosis or work programming. The present work describes the preliminary results of a research effort aimed at investigating the potential of ANNs in the interpretation of data from Nondestructive Testing (NDT), in special ultrasonic tests.

The increasing interest in the use of ANNs can be justified by the successful implementation experiences recorded in different areas. The method has been used in economics, medical and technical research, geology, physics and other fields, for solving and classification problems. The successes obtained derive from some of the very interesting intrinsic properties of a neural net, such as the possibility of non-linear modeling, and from the simple architecture that favorably distinguish them from other analysis methods [2].

An ANN works as a solid massive parallel processor, which is constituted by several simple units and has a natural propensity to store experimental knowledge and use it to create non-linear relationships between inputs and outputs [3]. In other words, an ANN is a highly interconnected network made of many simple processors. Each processor in the network maintains only one piece of dynamic information and is capable of only a few simple computations. An ANN performs computations by propagating changes in activation between the processors [4]. Using the ANN we can acquire, store and use the knowledge extracted from experts or experiments. The knowledge is kept in a steady state net of relationships between individual neurons and can be updated automatically using some kind of learning algorithm.

A net contains many paths, which are activated, to a certain degree, by the input vector. The signals generated are propagated and combined trough the various layers of the ANN, stimulating the various neurons, and finally generating the output signals [1].

The basic anatomy of an ANN could be divided into seven basic parts [5]:

- the set of individual processing units, or neurons;
- the state of activation of a processing unit;
- the function used to compute the output of a processing unit;
- the pattern of connectivity among the processing units;
- the rule of propagation employed;
- the activation function for each individual processing unit;
- the rules of learning that allow the determination of the pattern of connectivity between processing units.

The coefficients of correlation that establish the connectivity between neurons are called synaptic weights. Figure 1 shows the basic working mechanism of a neuron. It receives a series from inputs, each one carrying a specific synaptic weight. The result is filtered by an activation function that generates an output signal with certain intensity and that will serve as the stimulus for the next neuron.

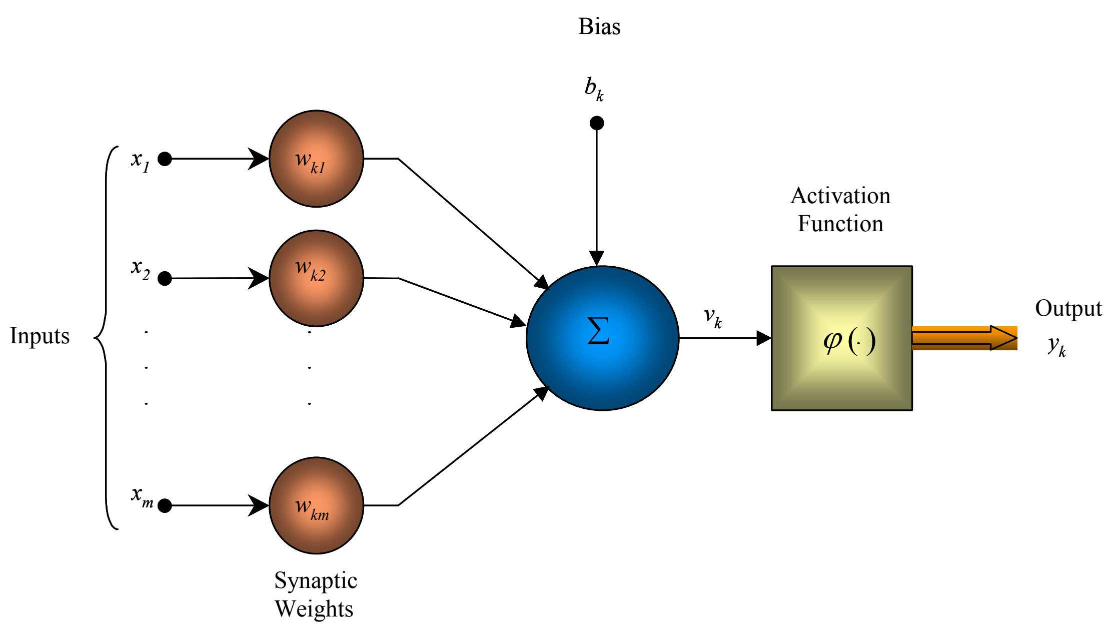


Figure 1 – A single neuron model [3].

One of the most interesting aspects is that, inside the neural network, it is possible to store and use knowledge extracted from experts or experiments, creating a kind of knowledge based system (KBS). The relationship map establishes how each link works. By using a multilayer perceptron (MLP), the present work aims to generate a non-linear correlation model between known input data, such as ultrasonic readings and material properties, with a certain output, compressive strength. For that purpose a net with an error back-propagation algorithm was created and exposed to a set of known results, in order to determine the estimation coefficients for each neuron on each layer. This report contains the first findings of this research and serves to demonstrate the great potential of using a back propagation algorithm to create an ANN for interpreting ultrasonic readings of concrete. The preliminary results are very encouraging, since they give clear indication that the models are robust and more accurate that traditional regression models. The major aim of the work is to test, explore and demonstrate the potential of artificial neural nets as tools for diagnosis, training and storage of non-structured knowledge in the civil engineering field.

Results: In this work, various architectures of ANNs, with three and four layers, were tested. The differences were created varying the number of neurons in the intermediary hidden layers, searching for equilibrium between estimate accuracy and computational cost. Different net configurations were tested, with up to 16 neurons in the first intermediary layer and 48 neurons in the second intermediary layer. The input parameters included several important concrete characteristics, such as the ultrasonic pulse velocity measurements (in m/s); water/cement ratio; cement type; aggregate type and size distribution; mix proportions and age (in days). The output parameter was the compressive strength, in MPa. The computational experiments were carried out using the Neural Network Toolbox of the MATLAB 5.3 software. For training the net it was used the data from 812 specimens that were subjected to measurements of ultrasonic pulse velocities and later subjected to compressive strength tests.

Each net was trained using the back-propagation algorithm, which tries to minimize the mean square error between the network output and the corresponding target values. Each training iteration is normally called an epoch. The training was limited on 10.000 epochs. After each iteration, the network explores the error surface searching for the greater gradient of reduction in the mean square error. The weights and biases are then adjusted to decrease the error. The initial weights and biases for each neural network were generated automatically by the program. This strategy allowed the exploration of different regions of the error surface.

Figures 2 and 3 show the evolution of the training of two nets. It can be noticed that the adjustment of the synaptic weights was quicker in the smaller net, with the mean square error dropping sharply until it reached the maximum value acceptable, defined by the user. It is interesting to observe that, like occurred in this case, the performance sometimes is not improved when the number of neurons is increased. For this reason, it is interesting to test the net several times if a solution isn't found on the first training exercise.

Figures 4 to 7 contain the plotting of the original data (the red diamonds), an estimation based on traditional statistical tools (the blue crosses) and the results of one some neural networks (the green circles). The first two were carried out with a subset of 130 data points, relative to just one type of cement and one type of aggregate. The other two were trained with all 812 available data points. In the larger nets it was considered adequate to increase the number of the hidden layers to two so the model would have more flexibility and be able to achieve a better estimate. According to the literature, using a larger number of hidden nodes can potentially improve the accuracy and convergence of the back-propagation algorithm at the cost of increasing the computational processing time [6]. As can be easily noticed, the neural networks usually fit the experimental data with better accuracy than traditional statistical models.

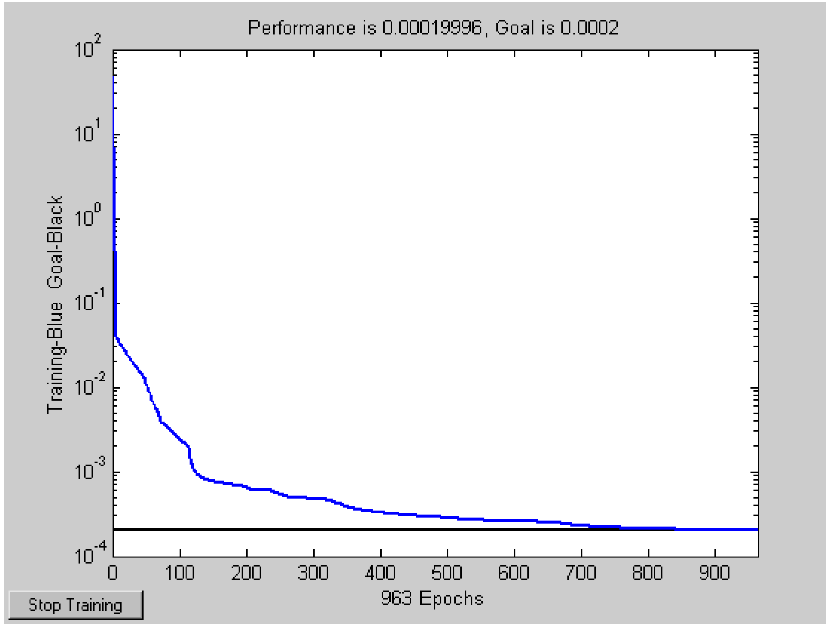


Figure 2 - Plotting of training for 963 epochs. – Net type: 3x8x48x1.

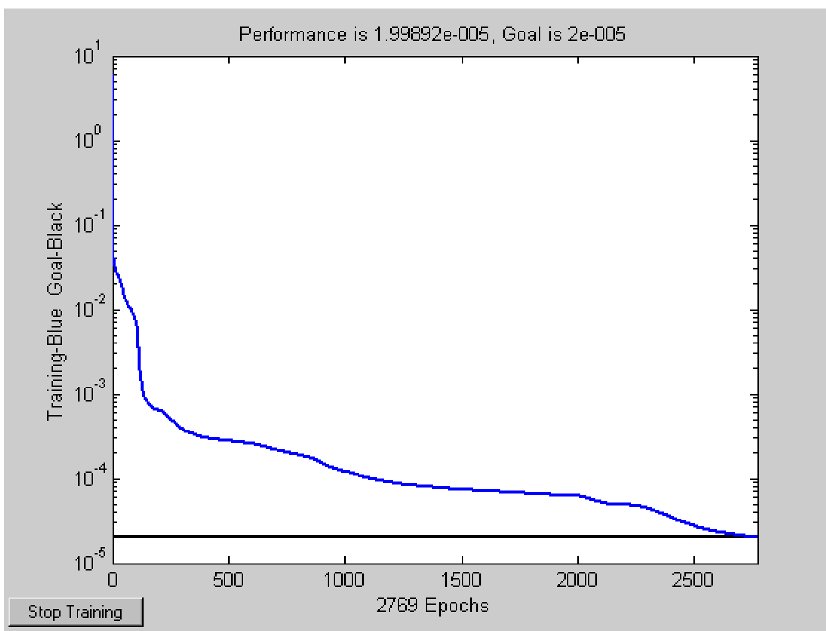
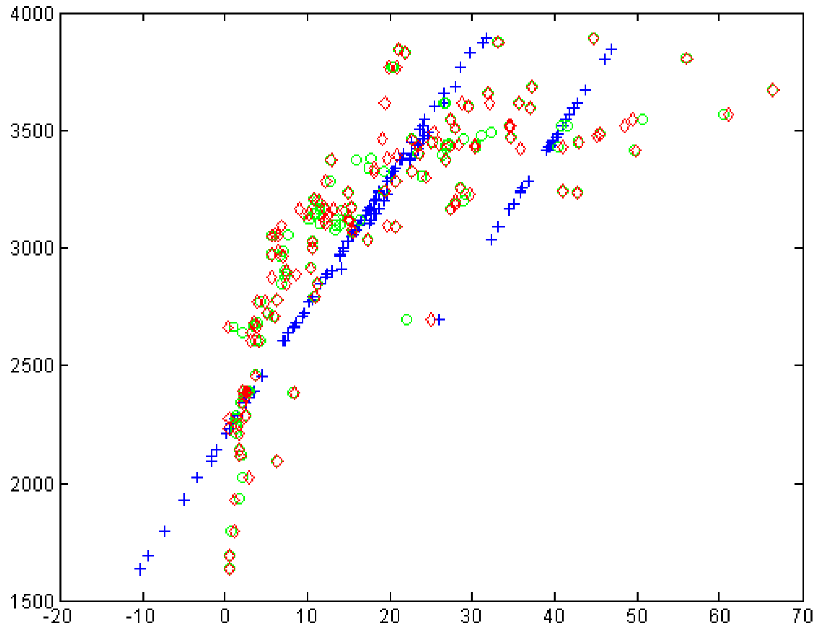
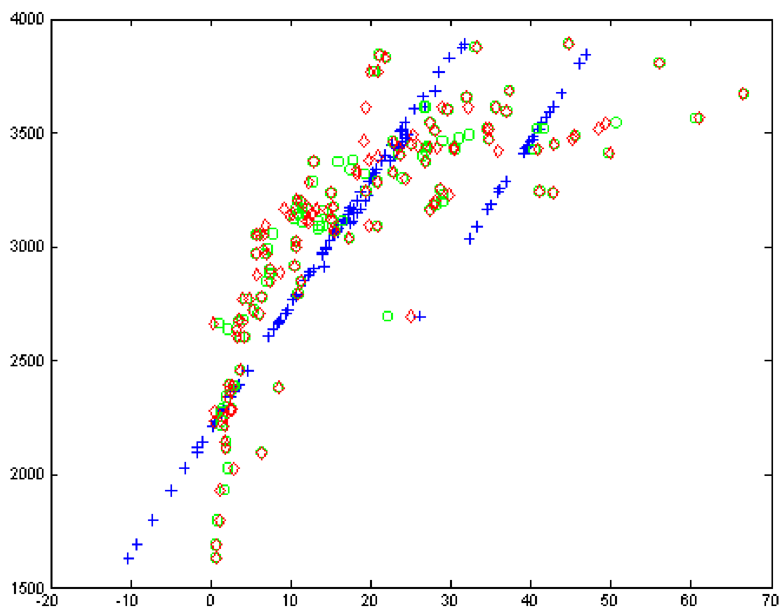


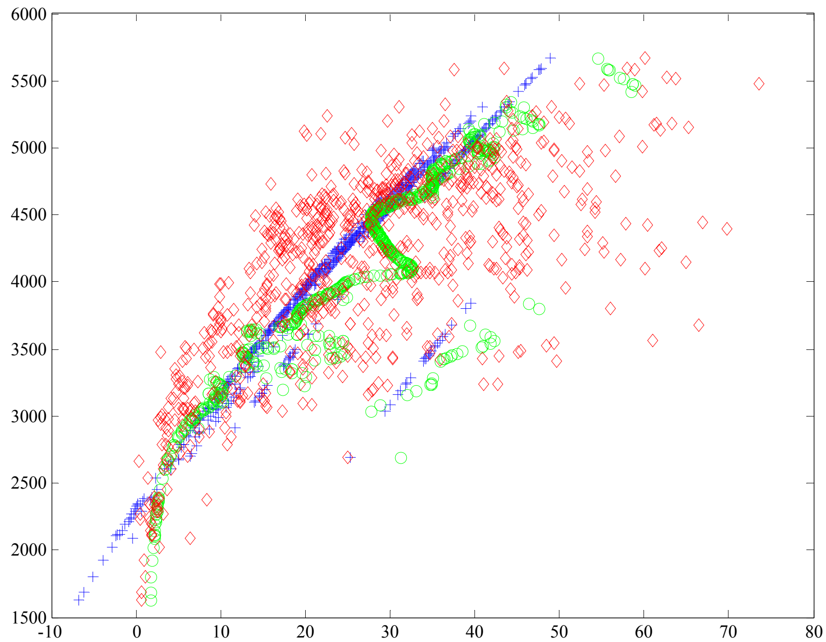
Figure 3 - Plotting of training for 2769 epochs. – Net type: 3x16x48x1.



Picture 4 - Plotting of ultrasonic vs. compressive strength – Net type: 3x8x48x1.



Picture 5 - Plotting of ultrasonic vs. compressive strength – Net type: 3x8x48x1.



Picture 6 - Plotting of ultrasonic vs. compressive strength – Net type: 3x4x16x16x1.

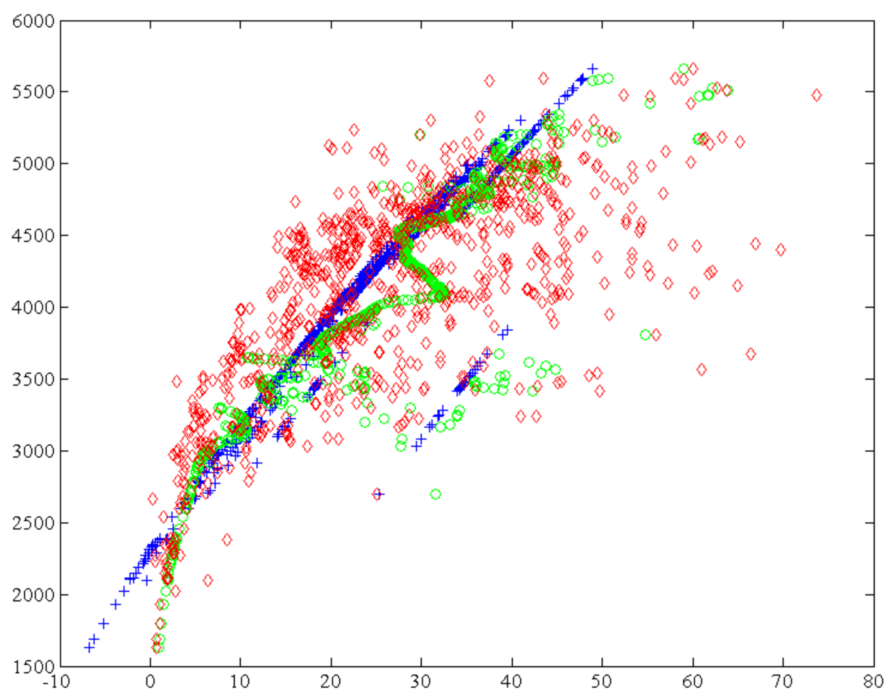
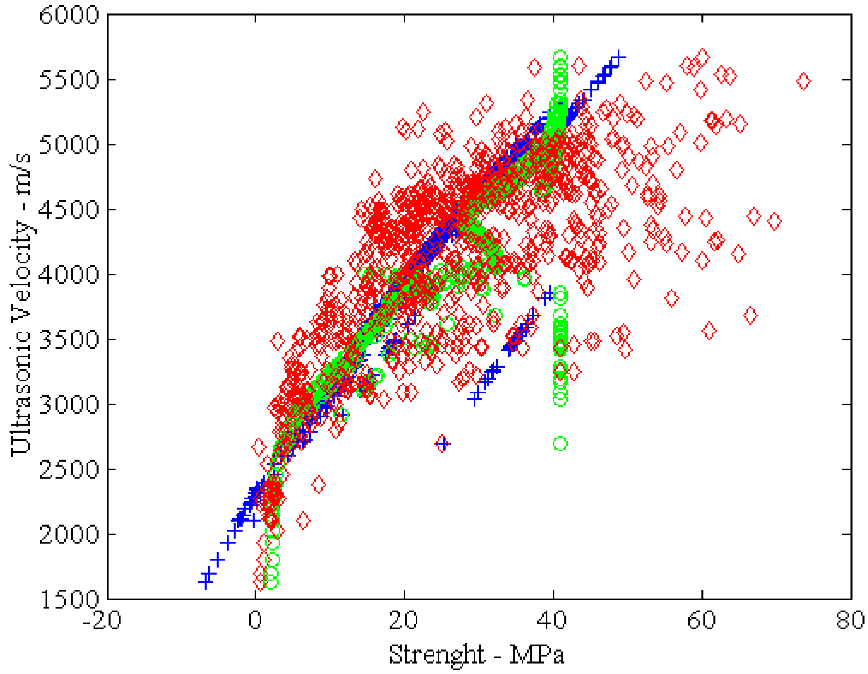


Figure 7 - Plotting of ultrasonic vs. compressive strength – Net type: 3x2x16x16x1.



Picture 8 - Plotting of ultrasonic vs. compressive strength – Net type: 3x4x24x24x1.

To avoid problems caused by the different order of magnitude of the input variables, all of them were normalized in the range of (0–1). The activation functions used were the hyperbolic tangent sigmoid transfer function in the hidden layers and the linear transfer function in the output layer.

Discussion: This work presents the preliminary results obtained using simple neural nets with just three input parameters: ultrasonic pulse velocity, concrete strength and age. Table 1 shows the performance of each of the different configurations tested. For a training period of 10.000 epochs, the performance goal, expressed by a minimum square error (MSE) of 4 MPa, was only achieved using the maximum number of neurons and a sub-set of the data. In this case, the goal was achieved in just 6.989 epochs. However, when the error condition was changed to 1 MPa, no net was able to achieve the goal in 10.000 epochs. When all the data available was used in the simulations the error grew, since the concrete specimens are very different, collaborating to increase the variability. It was interesting to observe that the performance sometimes is not improved when the number of neurons is increased. This occurs because the net explores the error surface using a different initial array of synaptic weights and the search for a solution evolves in the direction of the maximum error decrease. Therefore, a local minimum might be taken as the solution. For this reason, it is interesting to test the net several times, with different initial estimates of the synaptic weights, if a satisfactory solution is not found on the first training exercise.

Table 1 - Performance of different net configurations

ANN	Training Epochs	Maximum Error (%)
3x2x10x1	10000	2,29
3x2x24x1	10000	0,87
3x4x16x1	10000	0,87
3x4x24x1	10000	0,92
3x8x48x1	6986	0,25
3x2x16x16x1	10000	2,65
3x4x16x16x1	10000	1,84
3x4x24x24x1	5899	1,67

Conclusions: Simulation is a widely accepted tool in systems design and analysis. Because its basic concepts are easily understood, it has become a powerful decision-making instrument. The preliminary tests discussed in this paper clearly indicate that, with just three input parameters and three hidden layers, the estimation power of an ANN is already quite significant. The results have shown that an ANN is capable of modeling the relationship *strength vs. ultrasonic pulse velocity*. The precision of the estimates will depend on the quality of the information used to train the network. Increasing the number of neurons can also help to improve the model. However, when the size of the net grows or when the error criterion is tightened, the computational time needed to produce a result quickly increases. In short, the simulations carried out, using real data from ultrasonic tests performed in concrete, demonstrated that ANN can be very useful tools for interpreting the results of NDT. It is possible to create flexible and non-linear models that have better adherence to experimental data than traditional models. Moreover, it is possible to acquire and store knowledge in a dynamic configuration, creating models that can be constantly updated for different situations.

References:

[1] ENGEL, P. M., 2002, *Neural Networks*. Programa de Pós Graduação em Computação, Universidade Federal do Rio Grande do Sul. Porto Alegre.

[2] KHANDETSKY, V. & ANTONYUK, I. 2002, Signal Processing in Defect Detection using Back-propagation Neural Networks. In *NDT&E International*, 35: 483-488.

[3] HAYKIN, S., 2001. *Redes Neurais: Princípios e prática*. Trad. de Paulo Martins Engel. Porto Alegre: Bookman, 900p.

[4] RAJASEKARAN, S., FEBIN, M. F., RAMASAMY, J. V. 2002. Artificial Fuzzy Neural Networks in Civil Engineering. *Computers & Structures*, vol. 61, n. 2, p. 291-302.

[5] SRIRAM, R. D., *Intelligent Systems for Engineering: a knowledge-based approach*. Londres: Springer-Verlag, 1997, 804p.

[6] DHARIA, A., ADELI, H. 2003. Neural Network model for rapid forecasting of freeway link travel time. *Engineering Applications of Artificial Intelligence*, vol. 16, n. 6, p. 607-613.