

# ON THE USE OF ARTIFICIAL NEURAL NETWORKS IN THE ESTIMATION OF THE RESIDUAL LIFE OF FATIGUE-CRACKED SHEET SAMPLES

**Eliane Figueiredo Leite**

Departamento Básico, Faculdade de Engenharia Química de Lorena (FAENQUIL), Cx Postal 116, Lorena/SP  
e-mail: lilyleite@alunos.faenquil.br

**Adriano Francisco Siqueira**

Departamento Básico, Faculdade de Engenharia Química de Lorena (FAENQUIL), Cx Postal 116, Lorena/SP  
e-mail: adriano@debas.faenquil.br

**Carlos Antonio Reis Pereira Baptista**

Departamento de Engenharia de Materiais, Faculdade de Engenharia Química de Lorena (FAENQUIL), Cx Postal 116, Lorena/SP  
e-mail: baptista@demar.faenquil.br

**Abstract.** *The aim of the present work is to evaluate the fatigue crack behavior in commercially pure titanium sheet samples using Artificial Neural Networks (ANNs) coupled to a predictive damage mechanics model. Various ANN's architectures were tested and the best one, named NNI, consisted of an input layer with 6 neurons, a hidden layer with 15 neurons and an output layer with one variable representing the residual life. The training step was performed using 16 fatigue crack growth curves, obtained from center-cracked test pieces cyclic loaded with two load ratios:  $R = 0.1$  and  $R = 0.5$ . The input parameters were randomly selected from the initial normalized crack growth data. The predictions of NNI for residual life were performed for 14 curves "crack size versus number of cycles" that weren't previously used during the training step. The results showed an average correlation of 0.9974. The use of NNI coupled to the predictive model was able to improve the residual life estimations, which became closer to experimental data. A preliminary study on the necessary sample size for the neural network generalization was also performed. Through the evaluation of the synapse weights, together with the presentation of a theoretical result, an assumption for sample size was proposed for NNI.*

**Keywords:** *neural networks, fatigue crack propagation, modeling, titanium*

## 1. Introduction

Artificial neural networks (ANNs) are biological-inspired computer programs designed to simulate the way the information is processed by the brain. McCulloch and Pitts (1943), the precursors in this field, described a logical calculus of neural network in which neuron-physiology and mathematics were unified. ANNs are formed by hundreds of single units called artificial neurons or processing elements. The neural structure is the way these elements are arranged in layers and connected with coefficients or weights. ANNs build the knowledge by means of pattern detection and relations between the data used in their training. Thus, the results or output are obtained from previous experience and not by direct programming. ANNs are relatively easy to implement and can be used together with mechanistic models, acting in situations in which these become weak. The negative point is that the neural networks do not provide physical explanations for the obtained results, which are not completely interpretable (Shavlik, 1992). Nevertheless, ANNs have been increasingly used in the modeling of non-linear processes. Despite their biological inspiration, the architecture of ANNs evolved in a specific and independent form.

### 1.1. Basics on artificial neural networks

There are different types of ANN, but the basic principles are the same for all of them. The great advantage of ANNs is their ability to make use of information that is hidden in the input data (although they are not able to extract this information). The process of obtainment of the hidden information is called learning or training (Rumelhart, 1986). The multiple layer neural networks (MLNNs), trained with the retro propagating algorithm, are the most popular type of ANN. In a MLNN the first and the last layers are used for the input and output data, respectively. The remaining are called hidden layers. Each neuron in a particular layer is connected to every neuron in the next layer. The connection between neurons  $i$  and  $j$  is established by the weight  $w_{ij}$  and the threshold coefficient (bias)  $\delta_i$ . The output of the neuron,  $y_i$ , is determined by a transfer function  $g(u)$ , where  $u$  is given by Eq. (1). The most widely used transfer function, which is also adopted for the present work, is the sigmoid function, given by Eq. (2).

$$u = \delta_i + \sum w_{ij}x_i \quad (1)$$

$$y_i = g(u) = \frac{1}{1 + e^u} \quad (2)$$

During the training process, the difference between the desired and the calculated output data corresponding to a given input produces an error value that can be described in terms of the weights of the connections. These errors are retro-propagated through the network by an algorithm that adjusts the values of the weights in order to minimize the error function (Haykin, 2001). The objective is to obtain a trained network that is able to generalize its output. The classical retro propagating algorithm is described by Kovács (2002). According to Hagan (1995), the Levenberg-Marquardt method is one of the most efficient alternatives to the classical algorithm, being implemented in commercial languages like Matlab™. One major objective of the training process is to obtain a generalized ANN, i.e., a network that is able to give reasonable output for not previously known data. The basic condition for a good generalization is to use, in the training step, a representative and large enough input information, minimizing the need of extrapolations (Smith, 1993). There are various different formulae to determine the sample size, but according to Haykin (2001), it often occurs a large numerical difference between the sample size predicted by these methods and the size really required for the generalization. Another aspect to be taken in account is the effect of the input data on the network response. The method proposed by Garson (1991) to determine the relative importance of each input variable in the output data was employed in the present paper. This technique is based on the partitioning of the weights of the synaptic connections of the hidden and output layers in components related to each input neuron and provide an effective way to evaluate the influence of the input variables on the network behavior (Elgibaly and Elkamel, 1998; Faur-Brasquet and Le Cloirec, 2001; Grafton, 2001).

## 1.2. Neural networks and structural integrity

Being a powerful and versatile tool, ANNs are attractive for use in a wide range of engineering applications, including fracture mechanics and metal fatigue. Srinivasan *et al.* (2003) performed life predictions for high and low cycle fatigue in 316L stainless steel by adopting a neural network with 5 input parameters and variable normalization before the training step. Zenner *et al.* (2002) showed the ability of ANNs to perform fatigue calculations under random loading, avoiding the need of damage accumulation formulae. Fuji and co-workers (1996) modeled the fatigue crack propagation in nickel-based superalloys. Their ANN program was able to predict the effect of heat treatments on near-threshold crack growth rate. Venkatesh and Rack (1999) employed ANNs in high temperature fatigue life calculations for INCONEL 690 using 6 load parameters as input variables. Comparisons between ANN and Coffin-Manson equation results were done. Mohammed and Sudhakar (2000) evaluated the corrosion fatigue behavior of various steels by adopting the crack growth rate and stress intensity factor range as input variables. In all of the cited works the obtained results were considered quite satisfactory, thus qualifying the ANN-based procedures as an alternative to the conventional methods adopted in the mechanics of structural integrity.

In the present paper, a preliminary investigation on the use of neural networks to the estimation of residual fatigue life of center-crack titanium samples is performed. ANNs are employed with two main objectives: *i*) to compare and *ii*) to improve the results of a predictive method, described elsewhere (Baptista and Pastoukhov, 2003) and based on the fracture of damage accumulating volume elements along the crack path. The neural networks are used here to calculate the number of cycles spent by the fatigue crack to achieve a critical size, easily determined by static fracture mechanics. The complete description of the fatigue crack propagation still demands further development. Evaluations of the relative importance of the input variables, as well as some discussion on the sample size necessary to neural network generalization, are also done in our work.

## 2. Development of the work

In order to perform this work, various ANN architectures were tested. All the neural network development was done in Matlab™ environment, using the Levenberg-Marquardt algorithm in the training stage. The following sections are a step-by-step description of the developed work.

### 2.1. Input data

Experimental and numerical data sets were employed; both of them referring to titanium center-cracked panels submitted to stationary cyclic loading. The basic mechanical properties of the chosen material are the following: Yield strength = 349 MPa, ultimate tensile strength = 488 MPa, Young's modulus = 103 GPa, elongation to fracture = 27%. The experimental data set was collected from fatigue crack propagation tests of the titanium sheet (thickness = 1.5mm) samples (width = 50 mm, notch length = 12 mm) in a MTS servo-hydraulic machine. A sinusoidal loading (frequency = 10 Hz) was applied, the maximum load in a cycle being 8,0 kN. A number of 30 specimens were tested, 15 of them with a load ratio  $R = 0.1$  and the other 15 with  $R = 0.5$ . Initial crack measurements were made from a total crack length of  $2a_0 = 15$  mm. Crack size was measured using a traveling microscope (precision =  $\pm 0.01$  mm). The numerical data set

was provided by computer simulations of fatigue crack growth performed by a damage mechanics predictive model described elsewhere (Baptista and Pastoukhov, 2003). Again, a number of 30 curves (15 at each of the adopted load ratios) were generated and used in the present work. It should be clear that the predictive model does not need experimental crack growth data in order to perform the computer simulations. In the first stage of our work, only the experimental data were employed in the neural network development. After some trial-and-error investigation, an one-column array containing alternating values of crack size and number of fatigue cycles was adopted as the input variable set for the ANNs. These data were normalized before the training step (in order to maintain their values between 0 and 1), due to the saturation shown by the transfer function. Equation (3) was used for the variable normalization. In order to estimate the  $x_{max}$  values with 99% reliability, the Weibull probability distribution was employed. This distribution has shown the best fitting (Anderson-Darling coefficient  $AD = 1.4$ ) to the residual lifetimes when compared to other distributions: normal, exponential, lognormal and logistic. The minimum values for crack size and number of cycles were respectively 6.0mm and 3000 cycles, since no experimental datum was collected under these values.

$$x_t = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3)$$

## 2.2. Training the ANNs

The sigmoid transfer function, given by Eq. (2), was employed in the training stage. Since there is not a unique way to determine the network topology, various ANN architectures were tested and the best results in the training stage were obtained by the so-called NN1, a neural network with 6 input neurons, 15 hidden neurons and 1 output. Figure 1 shows a scheme of NN1. During the training, 16 data sets were supplied to NN1 (8 referring to  $R = 0.1$  and the remaining to  $R = 0.5$ ). These were obtained from the 6 initial points of the experimental crack growth curves, in the form of  $12 \times 1$  arrays containing the values of the number of cycles and the crack length. The desired output was the number of cycles for critical crack size achievement (i.e. the residual life).

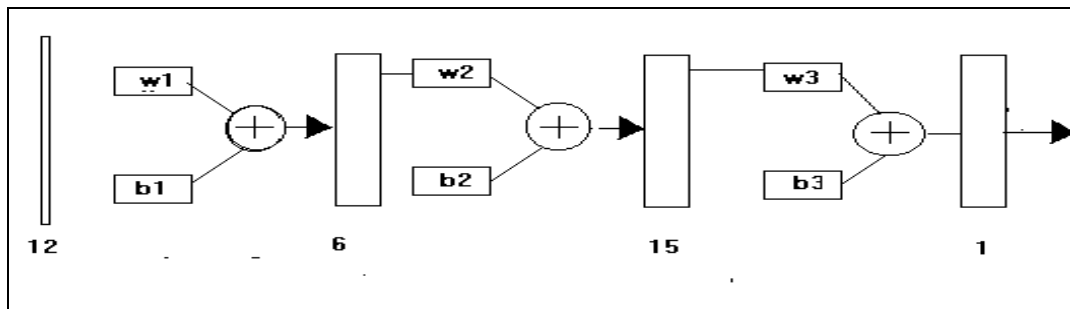


Figure 1. Architecture of NN1.

Since the purpose of a neural network is to codify the empirical knowledge (contained in the training sample) into a set of synaptic weights, it is then possible to analyze the information retained in these weights. To do so, the algorithm proposed by Garson (1991) and described by Eq. (4) was used in this work in order to determine the relative importance of each input variable in the output data set. In this equation,  $IR_{xp}$  is the relative importance of the input variable  $x_p$ ,  $n_e$  is the number of input variables,  $n_o$  is the number of neurons in the hidden layer,  $I_{pj}$  is the absolute value of the weight in the neural network, of the connection between the input variable  $p$  and the hidden layer  $j$ , and  $O_j$  is the absolute value of the weight in the neural network, of the connection between the hidden layer  $j$  and the output variable.

$$IR_{xp} = \frac{\sum_{j=1}^{n_o} \left[ \left( \frac{|I_{pj}|}{\sum_{k=1}^{n_e} |I_{pj,k}|} \right) |O_j| \right]}{\sum_{i=1}^{n_e} \left[ \sum_{j=1}^{n_o} \left( \frac{|I_{pi,j}|}{\sum_{k=1}^{n_e} |I_{pi,j,k}|} \right) |O_j| \right]} \quad (4)$$

### 2.3. The $\Delta K$ parameter

The purpose to investigate the effect of the input variables in the neural network results lead us to incorporate the stress intensity factor range ( $\Delta K$ ) as a new input parameter, in order to increase the available information and then improve the residual life predictions. This is one of the most important parameters of the linear elastic fracture mechanics, being largely used in semi-empirical models to correlate the fatigue crack growth rate ( $da/dN$ ). The  $\Delta K$  calculations were performed using Eq. (5), given by ASTM E647 standard test method for fatigue crack growth measurement. In Eq. (5),  $B$  is the specimen's thickness and  $W$  is its width;  $P_{max}$  is the maximum load in the cycle and  $a$  is the crack length. During the neural network development, two alternatives were tested: the first one was the construction of a network using the  $\Delta K$  parameter exclusively and in the second this parameter was included together with the crack size data. This comparison was necessary, since it is known that the  $K$  value depends on the crack length as well as the applied loading. As a result, two new networks were build: NN2 and NN3. Their results were compared to those given by NN1.

$$\Delta K = (1 - R) \frac{P_{max}}{B} \sqrt{\frac{\pi a}{2W^2} \sec\left(\frac{\pi a}{W}\right)} \quad (5)$$

### 2.4. Preliminary investigation on the sample size

The amount of input information necessary for satisfactory generalization of neural networks is an important area of research. In the present work an investigation is performed based on the synaptic weights of NN1. Since their values depend on the samples used during the training stage, and these samples are randomly obtained, the weights are considered as random variables. Defining  $n$  as the sample size adopted for the training stage and taking increasing values for  $n$  starting with  $n = 10$  (which means 10 experimental curves), successive neural networks were build based on NN1. For  $n = 10$ , the resulting synaptic weight matrix was denoted by  $w_1$ . For  $n = 11$ , we have a new weight matrix  $w_2$  and so on, resulting in various weight matrices  $w_i$ ,  $i = 1, 2, \dots, 10$ . Taking the sequence of weight matrices, we calculated the weight absolute increment matrices, denoted by  $d_i$ ,  $i = 1, 2, \dots, 9$  and defined by Eq. (6). Finally, the variations of the weight increments, denoted by  $v_i$ ,  $i = 1, 2, \dots, 8$ , were calculated according to Eq. (7). This equation gives the variation of the synaptic weight increments when the training sample size is increased and will be used together with a theoretical result in the formulation of a conjecture related to the sample size.

$$d_i = |w_{i+1} - w_i| \quad (6)$$

$$v_i = d_{i+1} - d_i \quad (7)$$

## 3. Results and discussion

In order calculate the residual life using NN1, we first employed 14 data sets that were not previously used in the training stage. In Figure 2 the normalized residual lifetime predicted by NN1 for these data sets is compared to the experimental results referred to load ratios  $R = 0.1$  and  $R = 0.5$  (7 curves for each loading regime). As can be seen in Fig. 2, NN1 was able to predict the residual life using only the information “crack size *versus* number of cycles” contained in the initial data points of crack propagation tests. The obtained predictions presented an average correlation factor of 0.9974.

The results concerning to the relative importance of each input variable are shown in Tab. 1. It can be seen that the first point is the most important for residual life predictions. This means that the initial crack growth rate measured in a test is determinant to the specimen fatigue life, i.e., if a test started out slow, it tended to remain slow for most of the test. This result is in accordance to the observations made by Virkler (1979) in his experiments of fatigue crack propagation.

As for the neural networks using  $\Delta K$ , the best architecture for NN2 has 2 input neurons, 3 hidden neurons and 1 output neuron. The architecture of NN3 is the same of NN1, except for the input array, whose size is  $18 \times 1$ . One interesting point is that NN2 is able to perform the residual life predictions using a lower number of neurons than NN1. The results obtained by NN2 for the same data sets used to evaluate NN1 are shown in Fig. 3. The mean quadratic error value for NN2 was 0.9588. The NN3 results are quite similar and showed an error coefficient of 0.9677. It is known that the neural network architecture reflects the amount of information contained in the training data set. In NN3, the number of neurons in the hidden layer is the same of NN1. These facts indicate that the information increment brought by  $\Delta K$  parameter is not significant or, in other words, the information carried by  $\Delta K$  is already included in the

experimental raw data. Thus we can conclude that NN is able to get all the information included in  $\Delta K$  parameter, which can be disregarded in the network calculations.

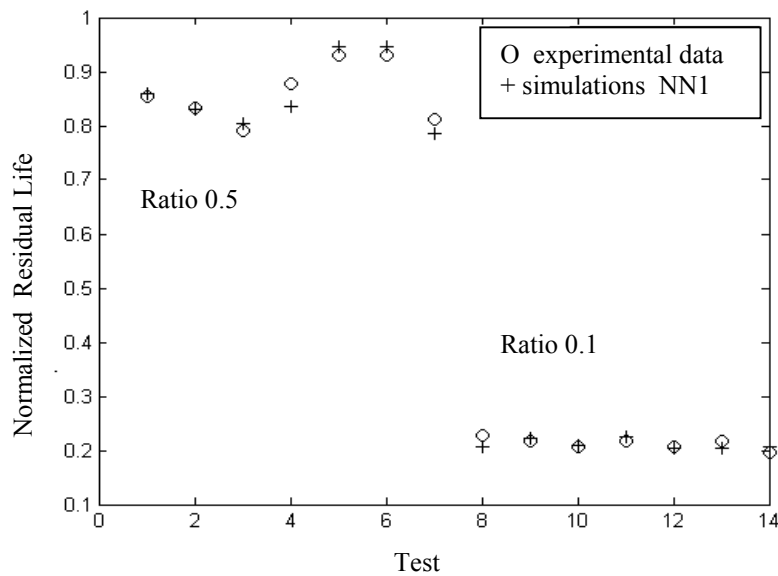


Figure 2. Comparison of experimental and calculated (NN1) normalized residual lives for two load ratios.

Table 1. Relative importance of the input variables.

Relative Importance (%)	Crack Length	Number of Cycles
1 <sup>st</sup> Point	11.53	20.35
2 <sup>nd</sup> Point	6.10	9.07
3 <sup>rd</sup> Point	8.21	9.83
4 <sup>th</sup> Point	3.73	4.74
5 <sup>th</sup> Point	7.67	4.97
6 <sup>th</sup> Point	9.59	4.20

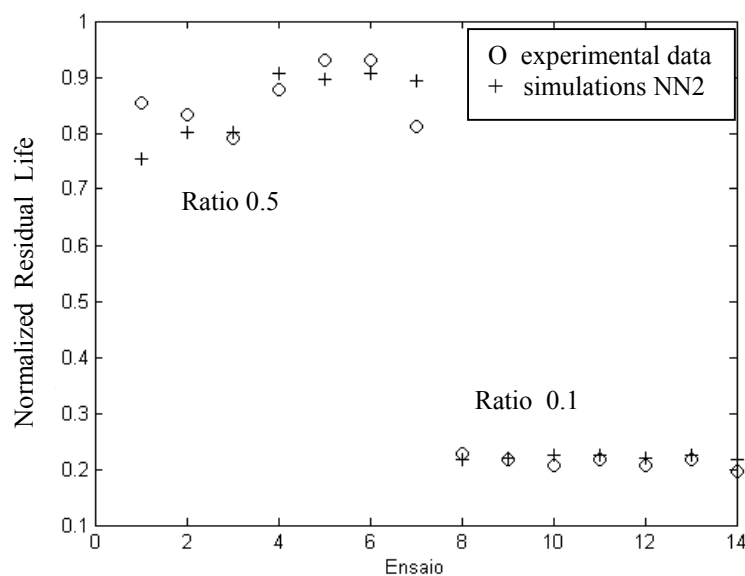


Figure 3. Comparison of experimental and calculated (NN2) normalized residual lives for two load ratios.

The next step is the employment of the neural network coupled to the damage mechanics predictive method in order to obtain residual life estimations for a wide range of loading conditions. Although being quite flexible, the predictive method is not completely successful in determining the residual life of the specimens with great precision. Thus, the data generated by the computer simulations of crack growth were introduced in NN1, trained with experimental data. The 6 input points were chosen by a randomization algorithm, which allowed to obtain 10 groups of 6 initial points for each set of 15 curves referred to each load ratio (0.1 and 0.5). The average result for each curve is plotted in Figures 4 and 5, together with the experimental and the simulated results. It is clear from these figures that the use of neural networks coupled with the predictive model improved the residual life results, which means that the predictions become closer to the experiments.

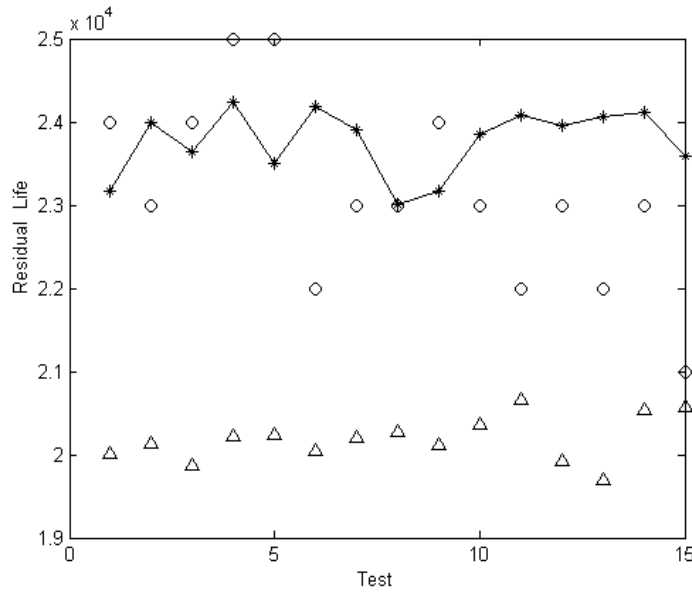


Figure 4. Comparison of experimental (O), simulated (Δ) and average neural network results (\*) for R = 0.1.

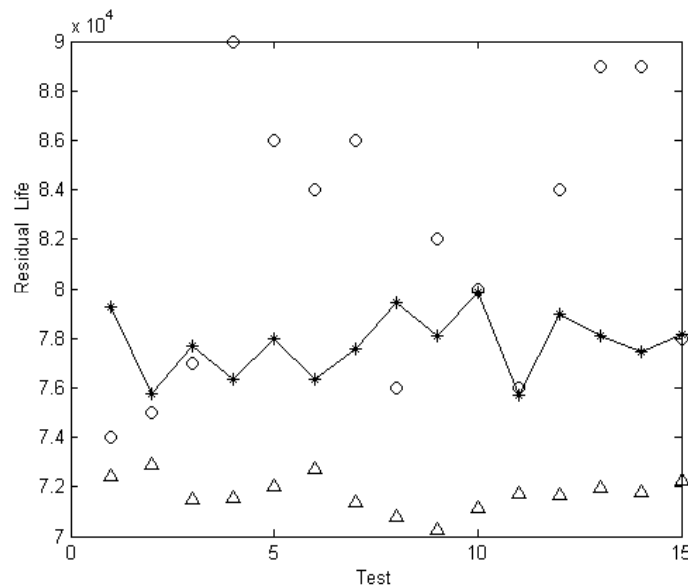


Figure 5. Comparison of experimental (O), simulated (Δ) and average neural network results (\*) for R = 0.5.

Another group of predictions was performed considering the load ratio R = 0.2 (for which no experimental data was collected). The results of coupled calculations are in Figure 6, together with the computer simulations. In this case, all the 10 results for each simulation are shown, in order to allow evaluating the scattering of the data. It can be seen that

the average neural network predictions (points linked by straight lines in Fig. 6) are above the experimental curves for  $R = 0.1$  and still far from  $R = 0.5$  values, which is a quite reasonable result. This procedure can be applied to any given  $R$ -value.

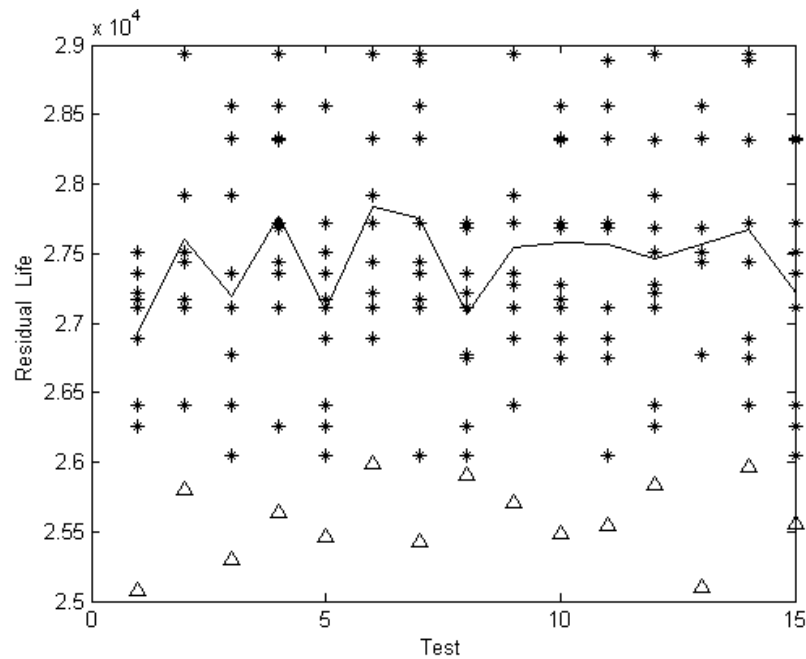


Figure 6. Comparison of simulated ( $\Delta$ ) and neural network results ( $*$ ) for  $R = 0.2$ .

In order to evaluate the variation of the synaptic weight increments of NN1 when the training sample size is varied, we took each of the  $v_i$  matrices obtained through Eq. (7) and calculated the arithmetic mean of its elements,  $E(v_i)$ . The obtained results for  $E(v_i)$  are given in Fig. 7, which shows that  $E(v_i)$  becomes negative for  $i \geq 6$ , corresponding to sample sizes above 15. A theoretical result that can be demonstrated states that the probability of the  $v_i$  components to be positive numbers suffer an exponential decay as  $n$  is increased. This result leads us to propose the following conjecture: “The training sample size is enough for network generalization when the average variation of the synaptic weight increments becomes negative”. This conjecture is in accordance with the best architecture obtained for NN1 and will be the object of further research by our group.

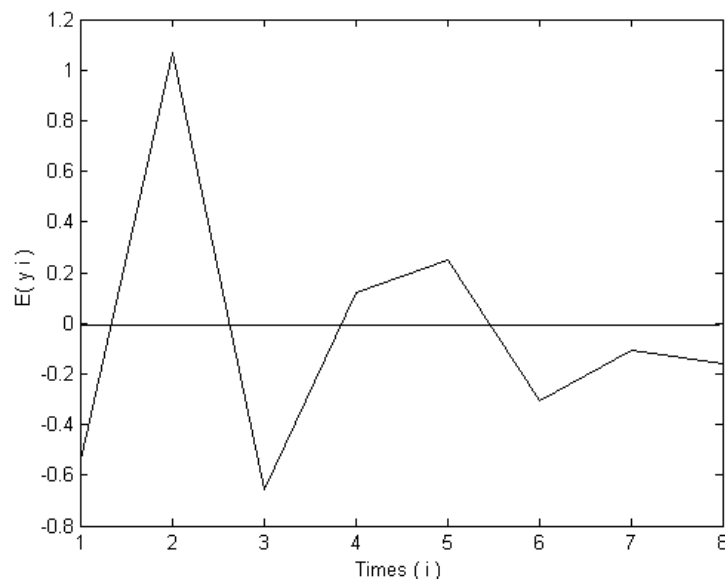


Figure 7. Mean values of the variation of the synaptic weight increments with sample size.

#### 4. Conclusion

In this work it was shown that artificial neural networks can be used in residual fatigue life calculations with reasonable results. It became evident that when the neural network is trained with experimental results, the information regarding the fatigue crack propagation is stored in its synaptic weights. Additional information contained in parameters such as  $\Delta K$  can be disregarded. When a group of specimens is tested under similar loading conditions, each one will generate a singular experimental curve, and this provides the scattering inherent to fatigue data. Since the neural network algorithm does not depend on the studied phenomenon, it can absorb the behavior patterns of the training sample and take in account this scattering when generating its predictions. On the contrary, the damage mechanics model used to simulate crack propagation results does not have this learning ability, resulting in residual life predictions systematically inferior to the experimental curves. When they are coupled, the neural network and predictive model can provide improved estimations of residual lifetime. This fact gives rise to a range of applications as wide as the ability of the predictive model in generating computer simulations of crack growth. The study of the effect of the synaptic weights lead to a conjecture about the sample size necessary to network generalization. Of course this subject still demands further research. Nevertheless, the preliminary results presented in this work allow one to establish the artificial neural networks as a promising tool in the assessment of fatigue crack behavior.

#### 5. Acknowledgements

The authors are thankful to FAPESP (proc. 04/00489-0).

#### 6. References

- Baptista, C.A.R.P., Pastoukhov, V.A., 2003, "A Damage Accumulation Model of Fatigue Crack Growth in Titanium", *Journal of the Mechanical Behavior of Materials*, Vol.14, pp.73-85.
- Elgibaly, A.A., Elkamel, A.M., 1998, "A new correlation for predicting hydrate formation conditions for various gas mixtures and inhibitors", *Fluid Phase Equilibria*, Vol.152, pp.23-42.
- Faur-Brasquet, C., Le Cloirec, P., 2001, "Neural Network Modeling of Organics Removal by Activated Carbon Cloths", *Journal of Environmental Engineering*, Vol.127, pp.889-894.
- Fuji, H., Mackay, D.J.C., Bhadeshia, H.K.D.H., 1996, "Bayesian Neural Network Analysis of Fatigue Crack Growth Rate in Nickel Base Superalloys", *University of Cambridge*, Vol.36, pp.1373-1382.
- Garson, G.D., 1991, "Interpreting Neural Network Connection Weights", *Artificial Intelligence Expert*, Vol.6, pp.47-51.
- Grafton, C., 2001, "Social Dimensions of Information Technology: Issues for the New Millenium", *Social Science Computer Review*, Vol.19, pp.121-125.
- Hagan, M.T., Menhaj, M.B., 1995, "Training Feed-Forward Networks with Marquardt Algorithm", *IEEE Trans. Neural Networks*, Vol.5, pp.989-993.
- Haykin, S., 2001, "Redes Neurais: Princípios e Práticas", Editora Bookman, Porto Alegre, Brazil.
- Kovács, Z.L., 2002, "Redes Neurais Artificiais: Fundamentos e Aplicações", Ed. Livraria da Física, São Paulo, Brazil.
- McCullock, W.S., Pitts, W.A., 1943, "A Logical of Ideas Immanent in Nervous Activity", *Bull. Math. Biophys.*, Vol.5, pp.115-133.
- Mohammed, E.H., Sudhakar, K.V., 2000, "Prediction of Corrosion-Fatigue Behavior of DP Steel through Artificial Neural Network", *International Journal of Fatigue*, Vol.23, pp.1-4.
- Rumelhardt, D.E., Hinton, G., Williams, R.W., 1986, "Parallel Distributed Processing: Explorations in the Microstructures of Cognition", MIT Press, Cambridge, USA.
- Shavlik, J.W., 1992, "A Framework for Combining Symbolic and Neural Learning", CSTR-92-1123, University of Wisconsin, available at [www.cs.wisc.edu/trs.html](http://www.cs.wisc.edu/trs.html), USA.
- Smith, M., 1993, "Neural Networks for Statistical Modeling", Van Nostrand Reinhold, New York, USA.
- Srinivasan, V.S., Valsan, M., Bhanu Sankara-Rao, K., Mannan, S.L., Raj, B., 2003, "Low Cycle Fatigue and Creep-Fatigue Interaction Behavior of 316L(N) Stainless Steel and Life Prediction by Artificial Neural Network Approach", *International Journal of Fatigue*, Vol.25, pp.1327-1338.
- Virkler, D.A., Hillberry, B.M., Goel, P.K., 1979, "The Statistical Nature of Fatigue Crack Propagation", *Transactions of the ASME*, Vol.101, pp.148-153.
- Zenner, H., Bukowski, L., Karkula, M., Marquardt, C., 2002, "Lifetime Prediction with the Application of Artificial Neural Networks", *Proceedings of 8<sup>th</sup> International Fatigue Congress*, Sockholm, Sweden, Vol.4, pp.2697-2704.

#### 7. Responsibility notice

The authors are the only responsible for the printed material included in this paper.