Available online at www.sciencedirect.com

ScienceDirect

Procedia Structural Integrity 3 (2017) 291–298

Structural Integrity Procedia

www.elsevier.com/locate/procedia

XXIV Italian Group of Fracture Conference, 1-3 March 2017, Urbino, Italy

Fatigue crack propagation in Ductile Cast Irons: an Artificial Neural Networks based model

Laura D'Agostino[a], Alberto De Santis[b], Vittorio Di Cocco[a], Daniela Iacoviello[b]*, Francesco Iacoviello[a]

[a]*Università di Cassino e del Lazio meridionale, DICeM, via G. Di Biasio 43, 03043, Cassino (FR) Italy*
[b]*Università di Roma "La Sapienza", DIAG, via Ariosto 25, 00185 Rome, Italy*

Abstract

All the available "Paris-like" models (analytical relationships between da/dN, crack growth rates, and ΔK, stress intensity factor amplitude) are not able to take into account the possible influence of all the parameters that influence the fatigue crack propagation process. Among them, the stress ratio R (e.g., K_{min}/K_{max}) is one of the most investigated and, although in the last decades the influence of R on the different propagation mechanisms has been widely investigated (e.g., crack closure effect), this parameter is often considered as an independent variable in the "Paris-like" models. A different approach can be followed using the Artificial Neural Networks that are able to consider all the possible parameters, with the condition of a satisfactory training stage. In this work, an artificial Neural Networks based model is optimized considering the influence of the stress ratio on the fatigue crack propagation in a ferritic-pearlitic Ductile Cast Iron.

Peer-review under responsibility of the Scientific Committee of IGF Ex-Co.

Keywords: Fatigue crack propagation; Artificial Neuran Networs; Ductile Cast Irons.

1. Introduction

Up to the first half of the last century, only malleable irons were able to partially offer a combination of grey iron castability and steel mechanical properties (first of all, toughness). These cast irons were obtained as a result of

* Corresponding author. Tel.: +39-677274061
E-mail address: iacoviello@dis.uniroma1.it

Peer-review under responsibility of the Scientific Committee of IGF Ex-Co.
10.1016/j.prostr.2017.04.048

extended annealing treatment of white iron, with a matrix microstructure that was characterized by different ferrite and pearlite volume fractions, as a function of the cooling cycle. The main problems of this procedure were the high costs and the difficulty to cast sound white iron components. In 1943, in the International Nickel Company Research Laboratory, a magnesium addition allowed to obtain a cast iron containing not flakes but nearly perfect graphite spheres. In 1948, at the American Foundryman Society Convention, it was announced that a small amount of cerium allowed to obtain the same result, Ward (1962), Labrecque (1998), Rundman (2016). After more than fifty years, ductile iron should be considered as a family of materials offering a wide range of properties depending on the chemical composition and heat treatment and the consequent microstructure modifications. Matrix microstructure importance is emphasized by the use of matrix names to commonly designate the different types of ductile irons (ferritic, ferritic-pearlitic, pearlitic, austenitic etc).

Ductile Cast Irons (DCIs) fatigue crack propagation resistance is strongly influenced by the matrix, by the graphite elements morphological peculiarities (shape, distribution and dimensions) and by the loading conditions, Iacoviello (2016). Focusing on the stress ratio parameter R, it strongly influences the crack closure effect, Elber (1971), that can be induced by the crack tip plasticity, by the presence of oxides on the fracture surface and by the fracture surface roughness. In DCIs the crack closure effect importance can be also enhanced by the presence of the graphite nodules that, near to the K_{min} values, can be an obstacle to the complete crack tip closure, Iacoviello (2008).

Many analytical models offer a relationship between the stress intensity factor amplitude, ΔK, and the crack growth rate, da/dN. Starting from the Paris law, Paris (1963), many other empirical or semi-empirical models were proposed in order to take into account the influence of the loading conditions (e.g., R values): Forman (1967), Yokobory (1969), Collipriest (1972) are among the oldest ones. The most evident limit of these models is their impossibility to describe all the possible influences of R on the da/dN-ΔK results, Iacoviello (2004). In fact, da/dN-DK interpolation curves obtained for different R values can have a "divergent", "parallel" or "convergent" behavior and the proposed models are not able to describe all these behaviors at the same time. A different approach that can consider the influence of different parameter is based on the Artificial Neural Networks (ANNs), and in this work a model based on ANNs is proposed in order to analyze the influence of the R value on the fatigue crack propagation in a ferritic-pearlitic DCI.

1.1. Artificial Neural Networks

Warren Mc Culloch (a neurophysiologist) and Walter Pits (a mathematician) first introduced the Artificial Neural Network (ANN) on 1943 to emulate the training and generalization processes of the human brain, Khanna (1990). In the years to come, the development of the computer performances boosted the use of the ANN in modeling and data analysis in many fields of application, where a traditional modeling based on constitutive equations was either not available or clumsy. Few examples can be the control of industrial plants and of complex systems in general, telecommunications, biology, Guyon (1990).

ANN has a structure that mimics the human brain one. The basic unit is the *neuron* (see fig. 1), each neuron receives different input signals x_i, $i = 1, 2, ..., n$ and performs a weighted sum with weights $w_i, i = 1,2,....,n$; if this sum is over a threshold value b (called *bias*), then an output signal y is generated

$$y = f\left(\sum_{i=1}^{n} w_i x_i - b\right) \tag{1}$$

otherwise the unit stays idle. Function f is called the neuron *activation function*; it is usually chosen as a *logsig* (see fig. 2)

$$f(t) = \frac{1}{1+e^{-t}} \tag{2}$$

to ensure a bounded output. Other choices are of course possible:

- *Step* function $f(t)=\begin{cases}0 & t<0\\ 1 & t\geq 0\end{cases}$
- *Linear* function $f(t)=mt+q$
- *Piece-wise linear* function $f(t)=\begin{cases}0 & t<t_{\min}\\ mt+q & t_{\min}\leq t\leq t_{\max}\\ 1 & t>t_{\max}\end{cases}$
- *Tansig* function $f(t)=\tanh(t)$
- *Radial basis* function $f(t)=\exp\left\{-\frac{(t-\mu)^2}{2\sigma^2}\right\}$

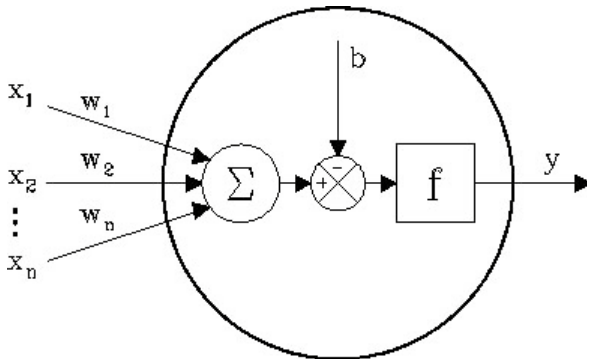

Fig.1. Neuron computational scheme.

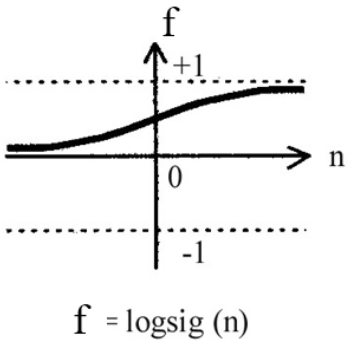

Fig.2. Sigmoid function.

The activation function formula (1) suggests that the structure of the computational scheme of the neuron depicted in fig.1 can be simplified by adding an extra input $x_{n+1}=1$ and an extra weight $w_{n+1}=-b$, obtaining the same value of *y*. This will be the case hereinafter.

An ANN is typically composed of an input layer, one or more intermediate or hidden layers, and an output layer. Fig. 3 shows a typical network with just one hidden layer. This particular structure is a feedforward network or a *multilayer perceptron*, where signals can travel only in one direction from the input layer to the output layer. In general each neuron in a hidden layer could be assigned a different activation function, but usually neurons in the same layer have the same activation function.

The weighted sum of the outputs of any layer is the input of the next layer. In fig.3, W_1 is a weight matrix of size $q\times n$, the *i-th* row contains the weights of the inputs $x_1,...,x_n$ to enter the *i-th* neuron of the hidden layer (the last weight is indeed the bias of the activation function of the *i-th* neuron); W_2 is a weight matrix of size $m\times q$, the *i-th* row contains the weights of the inputs $z_1,...,z_q$ to enter the *i-th* neuron of the output layer.

The setting of a feedforward ANN requires three steps

- *design*: given the number of inputs and the number of outputs, the number and size of the hidden layers is chosen; then the available input and output data are partitioned into two subsets of *training*, and *validation*; sometime a third set of *test* data may be required.
- *Training (learning)*: the weights W_1 and W_2 are modified so that the ANN outputs fit well the real outputs. This process is accomplished by minimizing the network fit error over the training set; as the input and output instances are defined by the experimenter, the training is said to be *supervised*.
- *generalization*: the ANN performance is evaluated over the validation set; if not satisfying, the design

step must be reconsidered to update the hidden layers structure, trying to avoid overfitting problems.

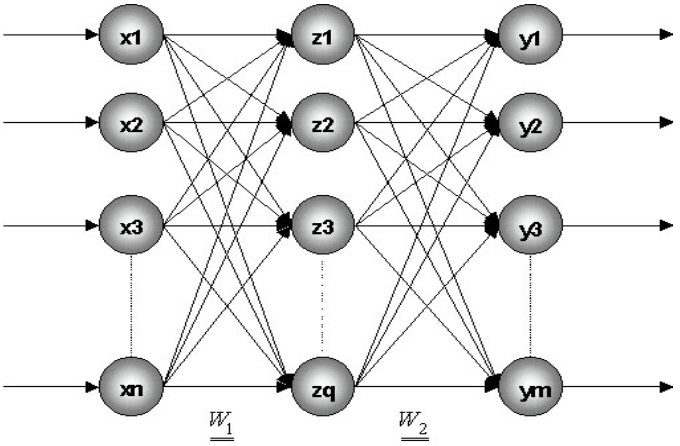

Fig. 3: Multilayer perceptron with one hidden layer

There is a large number of training algorithms whose basic task consist in adjusting the ANN weights so that a chosen loss function $L(e;W)$ of the ANN prediction error e and parameters W is optimized. Let e_t denote the ANN prediction error, i.e. the difference between the real output y_t and the network output $\hat{y}_t$, common loss functions are

- the error sum of squares $\mathrm{SSE} = \sum_t e_t^2$, or the square root version $\mathrm{SSEsq} = \sqrt{\sum_t e_t^2}$
- the sum of absolute errors $\mathrm{AE} = \sum_t |e_t|$
- the sum of percentage absolute errors $\mathrm{APE} = \sum_t \frac{|e_t|}{y_t}$

Therefore the training is accomplished by solving the following optimization problem

$$W^* = \arg\min L(e;W)$$

Any training algorithm recursively updates the weights value

$$W^{k+1} = W^k + \Delta W^k = W^k + \alpha_k d_k \tag{3}$$

thus generating a sequence of points converging to the minimum of the loss function. Vector d_k determines in the parameter space a decreasing *direction* of the loss function; it is usually taken as the loss function anti-gradient $-\partial L(e;W)/\partial W = -\nabla L(e;W)$ computed at $W = W^k$. Scalar α_k is the *step size* of the point update, and is responsible of the algorithm convergence rate. In the ANN framework, equation (3) goes by the name of *back propagation algorithm (BP)*, meaning that the updated weights W^{k+1} are fed back (propagated) into the network to compute new outputs to compare to the real ones; then a new value of the loss function gradient is computed at $W = W^{k+1}$ and by (3) a new update is obtained. The scalar α_k is called *learning rate*.

BP has some drawbacks: the rate of convergence strongly depends on the updating learning rates α_k (indeed, in the

basic ANN back propagation the learning rate is constant, $\alpha_k = \eta$), the algorithm may well get trapped in points of local minima. Therefore, many modifications to overcome these shortcomings have been proposed, a useful guide can be found in Saduf (2013). For instance, the Levenberg-Marquardt method (LM) consists in determining the parameter update sequence ΔW^k by solving the following equation

$$\left(J_k * J_k^T + \lambda * diag\left(J_k * J_k^T\right)\right)\Delta W^k = J_k * E_k \tag{4}$$

where J_k is the loss function gradient, E_k is the vector of the errors both computed at point W^k; $\lambda > 0$ is a damping parameter that can be dynamically adapted as well. The LM enforces larger displacements of the parameters update along the directions where the gradient is smaller, therefore avoiding the local minima trapping.

2. Investigated material. Experimental and numerical procedure

A ferritic – pearlitic DCI, with analogous ferrite and pearlite volume fractions, was investigated (chemical composition and some mechanical properties are shown in Table 1). The microstructure morphology showed a peculiar "bull's eye" morphology and graphite elements were characterized by a high nodularity level.

Table 1. Investigated ferritic-pearlitic DCI chemical composition (wt%) and mechanical properties.

C	Si	Mn	S	P	Cr	Mg	Sn
3.65	2.72	0.18	0.010	0.03	0.05	0.055	0.035
UTS [MPa]		YS [MPa]		A%		HB	
500		320		7%		180-230	

Fatigue tests were performed using 10 mm thick CT (Compact Type) specimens. According to E647 ASTM (2015), using a computer controlled (100 kN) servo-hydraulic testing machine in constant load amplitude conditions, with a sinusoidal waveform. Tests were performed in air at room temperature, with a loading frequency of 20 Hz, considering eight different stress ratio values (e.g. $R = P_{min}/P_{max}$ equal to $0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$). Crack lengths were measured using a compliance method with a double cantilever crack mouth gauge and were controlled using an optical method with a 40x magnification. Fracture surfaces were investigated by means of a scanning electron microscope (SEM).

Data for the ANN set up are therefore the 8 vectors $y_1, y_2, \cdots, y_8$ of the crack growth rate da/dN corresponding to the 8 different values of the stress ratio R. An ANN with radial basis activation function is able to approximate the mapping from R to da/dN. A radial basis network is a network with a hidden layer of radial basis neurons and an output layer of linear neurons. The procedure is implemented by the Matlab Neural Networks toolbox. The network is trained by selecting as input-output pair the following values

$IN = \begin{pmatrix} 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.7 & 0.8 \end{pmatrix}$, of size 1×7
$OUT = \begin{pmatrix} y_1 & y_2 & y_3 & y_4 & y_5 & y_7 & y_8 \end{pmatrix}$, of size 19×7

Here is the Matlab script for training

```
eg = 0.02;     % sum-squared error goal
sc = 0.08;     % spread constant σ for the activation function
net = newrb(IN,OUT,eg,sc);
```

The network is then applied to estimate the crack grow rate values for the stress ratio value of 0.6 which was not included in the training set

```
y6 = net(0.6);
```

The network output is also computed for a stress ratio of 0.45, an intermediate value between 0.5 and 0.6,

```
y4.5 = net(0.45).
```

3. Results and discussion

The results of the fatigue crack propagation tests are shown in Fig. 4. The influence of the stress ratio R on the fatigue crack propagation results is quite evident. SEM analysis of the fracture surfaces shows the graphite nodules-matrix debonding and the evident cleavage around the nodules, corresponding to the ferritic shields (Fig. 5 and 6, respectively).

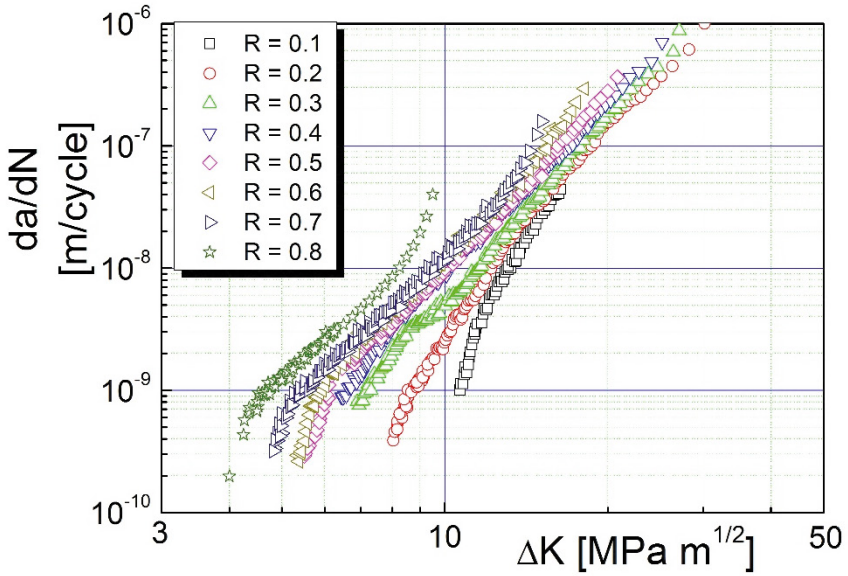

Fig. 4. Stress ratio influence on the fatigue crack propagation in a ferritic-pearlitic DCI.

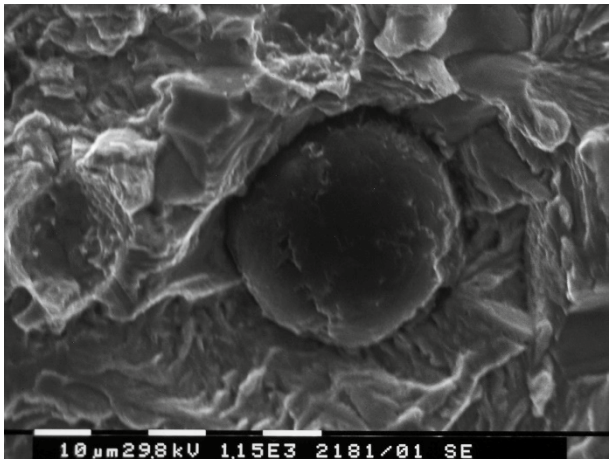

Fig. 5: Debonding (R = 0.1; ΔK = 11 MPa√m).

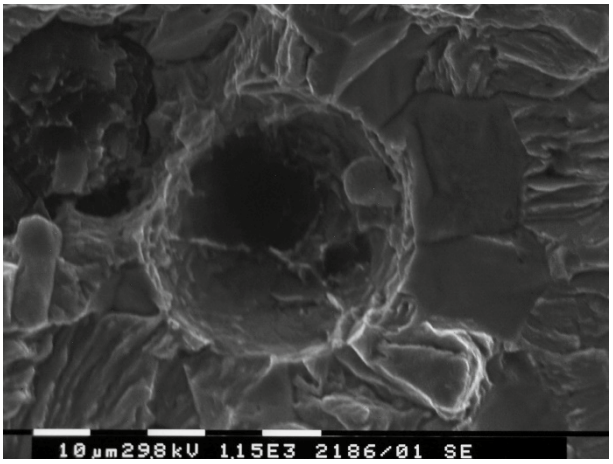

Fig. 6: Debonding (R = 0.1; ΔK = 16 MPa√m).

The ANN features an high predictive power. The output y6 well approximates the experimental values, as shown on

Fig. 7. Also the second trial gave a satisfactory result, the network output y4.5 yields values that are quite compatible with the experimental ones reported on Fig. 8.

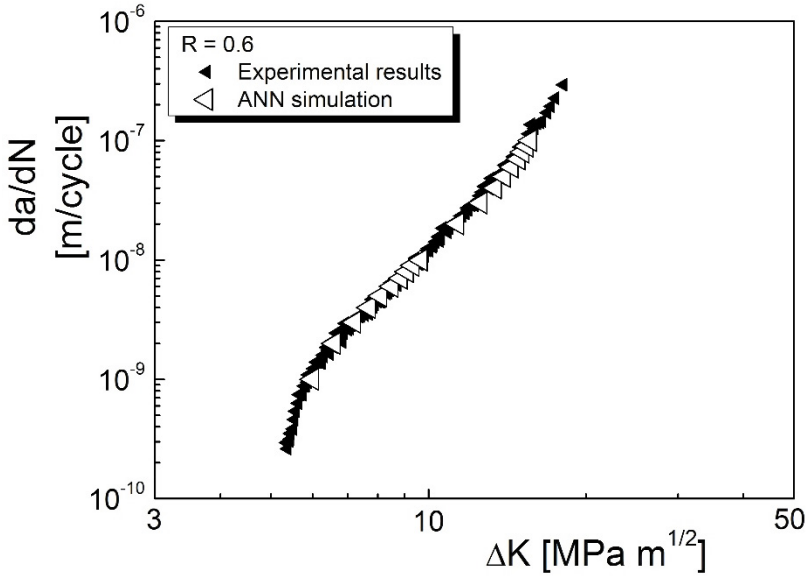

Fig. 7. Experimental and numerical results (R = 0.6).

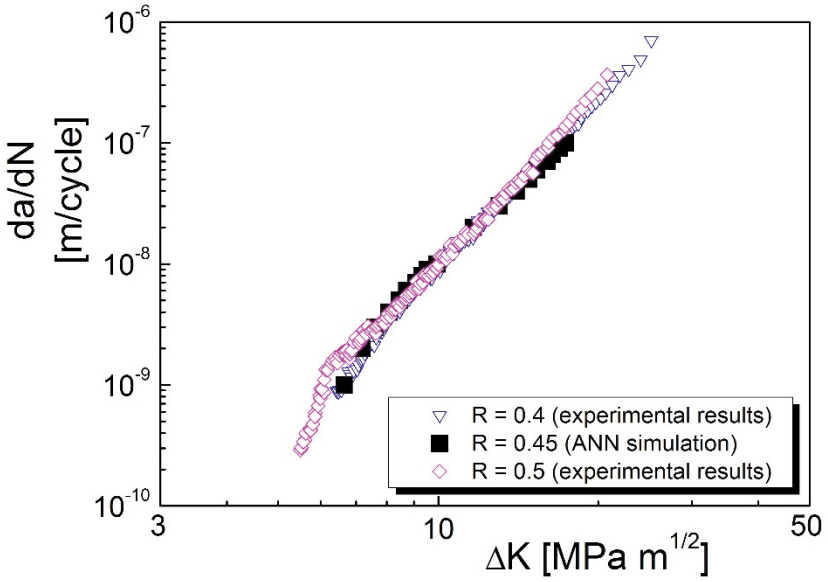

Fig. 8. Experimental (R = 0.4 and 0.5) and numerical (R = 0.45) results.

4. Conclusions

In this work the influence of the stress ratio on the fatigue crack propagation in a ferritic-pearlitic DCI was investigated. Eight different stress ratios were considered, ranging between 0,1 and 0,8, and an Artificial Neural

Network based model was developed in order to simulate the influence of the stress ratio on the da/dN-ΔK fatigue crack propagation results. Based on the experimental and numerical results, it is possible to summarize the following conclusions:

- stress ratio R influences the fatigue crack propagation in ferritic-pearlitic DCIs;
- among the main fatigue crack propagation micro-mechanisms, the most evident are the graphite nodules-matrix debonding and the ferritic shield cleavage;
- the numerical procedure based on the Artificial Neural Network "radial basis" is able to simulate satisfactorily the influence of the stress ratio on the fatigue crack propagation, as shown by the results in the da/dN-ΔK diagram on Figs. 7-8.

References

ASTM E647-15e1, 2015. Standard test method for measurement of fatigue crack growth rates.

Collipriest, J.E., 1972. An experimentalist's view of the surface flaw problem, ASME, 43.

Elber, W., 1971, Damage tolerance in aircraft structures. ASTM STP 486. Philadelphia (PA), American Society for Testing and Materials, 230.

Forman, R.G., Kearney, V.E., Engle, R.M., 1967. Numerical analysis of crack propagation in a cyclic-loaded structure. J. Basic Eng., ASME Trans., 89D, 459.

Guyon, I., 1990. Neural networks and Applications, Amsterdam, Computer Physics Reports.

Iacoviello, F., Iacoviello, D., Cavallini, M., 2004. Analysis of stress ratio effects on fatigue propagation in a sintered duplex steel by experimentation and artificial neural network approaches. Int. J. of Fatigue. 26, 819-828.

Iacoviello, F., Di Bartolomeo, O., Di Cocco, V., Piacente, V., 2008. Damaging micromechanisms in ferritic–pearlitic ductile cast irons. Mater Sci Engng, 478, 181–186.

Iacoviello, F., Di Cocco, V., Cavallini, M., 2016. Fatigue crack propagation and overload damaging micromechanisms in a ferritic–pearlitic ductile cast iron, Fatigue Fract Engng Mater Struct, 39, 999–1011

Khanna, T., 1990. Foundations of Neural Networks, Addison Wesley.

Labrecque, C., Gagne, M., 1998. Review ductile iron: fifty years of continuous development. Canadian Metallurgical Quarterly. 37, 343-378.

Mason, J.C., Ellacott, S.W., Anderson, I.J., 1997. Mathematics of neural networks: models, algorithms and applications, Boston: Kluwer Academic.

Paris, P., Erdogan, F., 1963. A critical analysis of crack propagation laws, Journal of Basic Engineering. Transactions of the American Society of Mechanical Engineers, 528-534.

Rundman, K.B., Iacoviello, F., 2016. Cast Irons, Reference Module in Materials Science and Materials Engineering, 1-11.

Saduf, M.A.W., 2013. Comparative Study of Back Propagation Learning Algorithms for Neural Networks. International Journal of Advanced Research in Computer Science and Software Engineering, 3:12, 1151-1156.

Ward, R.G., 1962. An introduction to the physical chemistry of iron and steel making. Arnold, London.

Yokobory, T., 1969. Physics of strength and plasticity, A.S. Argon Ed., M.I.T. Press, 327.