

## Adjustment in the Least Square Meaning of a Neural Model for Electric Arc Quenching in High Voltage Circuit Breakers

A. Ziani<sup>1</sup>, H. Moulai<sup>1</sup>

<sup>1</sup>Laboratory of Electrical and Industrial Systems,  
FEI, USTHB, BP 32 Bab Ezzouar, Algiers, 16311, Algeria.  
ziani08@yahoo.fr

**Abstract.** In this paper, a simulation of the breaking arc in high voltage circuit breakers based on a combination of the least square and the neural networks techniques will be presented. A simulation program has been developed by the means of Matlab software that enables to adapt polynomial functions to the neural simulations. The obtained results show the efficiency of the method on a 245kV/50kA line breaker.

**Key words:** Electric arc, High voltage, Neural networks.

### 1. Introduction

The monitoring and control of the phenomena related to electric arcs extinction in high voltage (HV) circuit breakers gets call more and more to artificial intelligence techniques [1, 2]. These enable to set up relationships between very complex physical parameters that are implicated at the contacts opening in order to ensure a better protection of electric networks. Moreover, in spite of the significant scientific works achieved on HV arcs extinction, as well in the theoretical domain than experimental, the physical phenomena remain difficult for analytical modeling that would enable to foresee their behavior with a weak margin of uncertainty [2-6].

Artificial Neural Networks (ANN) are the most used in complex problems for which there are many difficulties to achieve a large number of measurements and where the underlying laws aren't well known [7]. ANN appear therefore well indicated to respond to the need of simulation of breaking electric arcs.

For this effect, the goal of this work is first of all to conceive a neural network with the electric arc time of breaking as the input and the corresponding arc voltage as an output. This will enable us to dispose a set of data that will be regrouped according to least square criteria in order to better model the electric arc extinction.

Indeed, the least square method is often used to refine the developed models [1, 8-10] and to reduce the committed errors when estimating the different energies. Moreover, this method has been shown to be enough fruitful in the modeling of non-linear phenomena [9, 10]. A program developed under

Matlab environment enables us to optimize the different coefficients to approximate the arc currents and voltages.

In this paper, we will try to answer to the two following questions: which polynomial function is better adapted to arc current simulation? And which polynomial function can best model the arc voltage?

The simulation will be leaded in a line high voltage circuit breaker 245kV/50kA/50Hz with SF<sub>6</sub> as a breaking medium, between 0 and 90μs. An ANN with a training supervised with the Levenberg-Marquardt algorithm will be used in order to control the breaking of a default current of about 90% of the cutting capacity. This breaker has been the subject of several experimental studies, especially those achieved by Schavemaker et al [11] and J. L. Guardado [12].

### 2. Formulation of the Problem

By considering  $p$  the number of points  $(x_i, y_i)$  generated by the neural network such as:  $1 \leq i \leq p$ , one must search a polynomial curve  $y=f(x)$  of degree  $n$  that brings closer the points  $(y_i, x_i)$  such as the optimal coefficients of this polynomial are those that minimizes the average quadratic residue  $R$  of the interpolation. One can write therefore:

$$R = \frac{1}{p} \sum_{i=1}^p (y_i - f(x_i))^2 \quad (1)$$

Where  $f(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n \dots$  with  $a_0, a_1, \dots, a_n$  the polynomial coefficients.

Searching to find the best coefficients of the polynomial in the least square meaning returns to solve the following system of equations:

$$\begin{cases} \frac{\partial R}{\partial a_0} = 0 \\ \frac{\partial R}{\partial a_i} = 0 \quad \text{for } i \text{ between } 1 \text{ and } n \\ \frac{\partial R}{\partial a_n} = 0 \end{cases} \quad (2)$$

One obtains thus (n+1) equations where the (n+1) unknowns are the polynomial coefficients. The solving of this linear system enables to determine the polynomial of approximation of the experimental curve in the least square meaning. The choice of the polynomial degree is the most delicate step of the algorithm. Indeed, it conditions the smoothing quality: more this latter will be elevated, more the polynomial will have tendency to pass all over the points of the curve.

The "polyfit" function of Matlab software was used for the linear interpolation in the least square meaning. This function enables to minimize the value of R and to achieve the optimization of the coefficients of the polynomial function  $f(x)$ . In order to deduce the error between the experimental values  $V_i$  of the arc voltage and those obtained by the polyfit function model, we used the "polyval" function which returns the value of the interpolation polynomial for all values of time between 0 and 90  $\mu$ s. The error  $E_i$  committed at point  $x_i$  is then equal to:

$$E_i = V_i - \text{polyval}(x_i) \quad (3)$$

### 3. Presentation of the neural network

The Feed-forward propagation network is chosen to simulate the electric arc breaking (under this configuration, all neurons of a given layer are usually connected to all neurons of the following layer). These networks are very used in identification and modeling of non linear systems [1, 3, 13]. Indeed, they enable to approximate every continuous phenomenon with only one layer [2].

As the wanted output of the network is known, we must therefore execute a supervised training. Two vectors of data will be then provided to the network:

The input vector  $(t_0, t_1, \dots, t_j, \dots, t_n)$  and the output vector  $(v_0, v_1, \dots, v_i, \dots, v_n)$  where each point of the simulation is represented by  $(t_i, v_i)$  at the time  $t_i$  and corresponding voltage  $v_i$ .

These input-output of the neural model are respectively the experimental values of time and arc voltage of the considered breaker. The corresponding structure is represented in figure 1 [11, 12]. One can first observe a sharp increase of the arc voltage at the contacts opening that corresponds to the arc generation and then, relatively weak arc voltages with regard to the network voltage are measured.

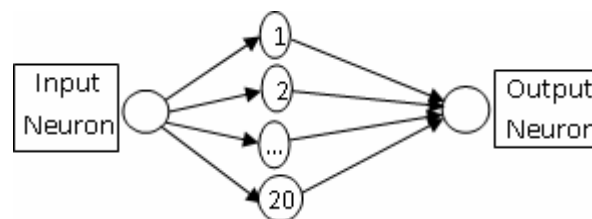


Fig. 1. Structure of the retained neural network

#### A. Learning Phase of the neural network

Although the choice of training algorithm in a feed-forward network is difficult, the Levenberg-Marquardt algorithm presents in general the fastest convergence for modeling of physical phenomena problems [7, 14]. In this work, we chose a "trainlm" type of training. For this type of training, predefined input vectors are provided to the ANN, and the goal of the network is to be able to determine in another input vector the simulated values.

The training phase is executed while adjusting the weights by the back-propagation algorithm until obtaining a quadratic error fixed in advance between the wanted output and the output of the neural network. For the different iterations, the back-propagation algorithm provides the MSE index of performance (Mean Square Error performance function). The weights will be initialized by uncertain manner.

Back-propagation algorithms that we used update the weights and the bias in the meaning to fast minimizing the MSE index of performance. The training will be considered as complete after 500 epochs (optimal number obtained by simulation) or when the maximal error, on all output neurons, between the wanted value and the real value is lower than  $10^{-5}$  for any given input of the training database.

#### B. Neural Network Programming

The network programming has been achieved under Matlab environment. It required 2 files: A first file presenting the training data under matrix form where the first line of the matrix includes the different values of electric arc breaking time in microseconds, and the second line represents the values of the arc voltage in Volts. The second configuration file gives the necessary informations about the ANN structure and the training parameters.

In this work, the "newff" Matlab command has been used to create the forward propagation ANN network.

#### C. Structure of the Network

The network is constituted from three layers:

- The input layer constituted by only one neuron.
- A hidden layer with 20 neurons having a sigmoid as a function of activation. The "logsig" command

enables to create this function under Matlab environment.

- An output layer constituted from one neuron where the function of activation type is positive linear and the Matlab command is designated as "poslin".

#### D. Testing Phase

For the testing phase, we compared the values simulated by ANN with those obtained experimentally by Schavemaker et al [11] for a period of extinction between 0 and 90 $\mu$ s. For that purpose, other data different from those used in the training phase have been used.

## 4. Results of simulation

All simulation phases of the neural network and the least square method have been executed under Matlab environment.

Figure 3 illustrates the performance of the used network. The training performance reached an error of about  $3.7 \cdot 10^{-13}$  at the end of 26 epochs. One remark that the network converges after 20 iterations and that the quadratic error average reaches the fixed value of  $10^{-5}$  after 25 epochs. This confirms the efficiency of the "lm" algorithm used for this modeling as well as its generalization.

Current simulation results are represented on figures 4 and 5. They show clearly the effect of the polynomial function degree on the arc current estimation. Weak gaps between the least square technique obtained values for degree 1 and the experimental values are found (figure 4). Furthermore, on figure 5, one notes that a polynomial of elevated degree will have tendency to produce inflection points and generates gaps at the end of breaking. A deviation of 300A is obtained at  $t=90\mu$ s. Figure 6 shows that polynomials of weak degree minimize better the approximation error of the current.

Figures 7, 8 and 9 present the results of arc voltage variations obtained respectively with polynomials of degrees 1, 5 and 10. It appears clearly that an elevated polynomial degree reproduces better the arc voltage evolvement. The arc voltage deviation reaches 400V at  $t = 40\mu$ s and 480V at  $t=88\mu$ s for a polynomial of order 1.

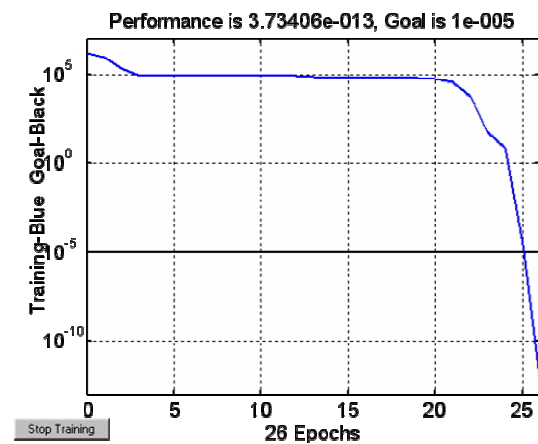


Fig. 2. Artificial Neural network performances

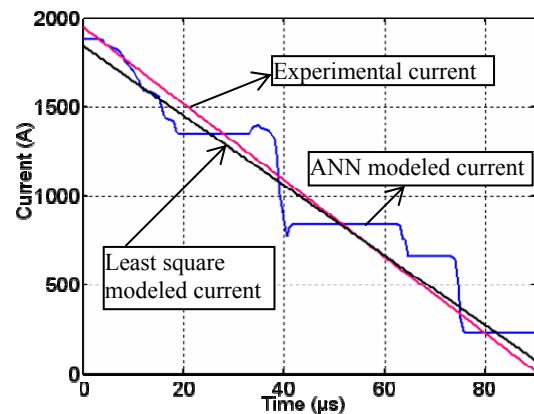


Fig. 3. Variations of the arc current as a function of time

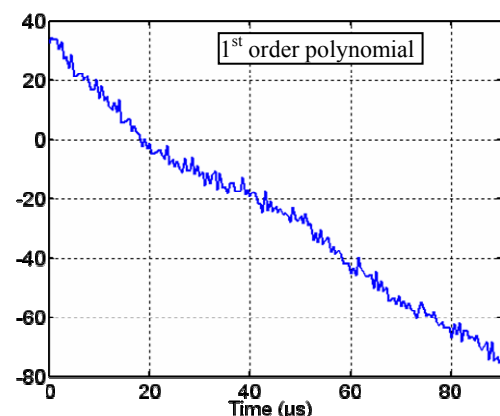


Fig. 4. Error variations as a function of time

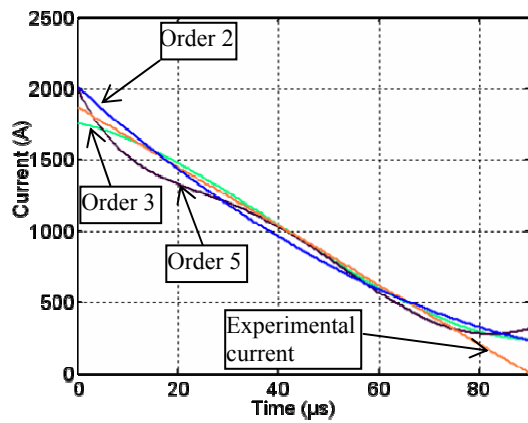


Fig. 5 . Arc current variations as a function of time.

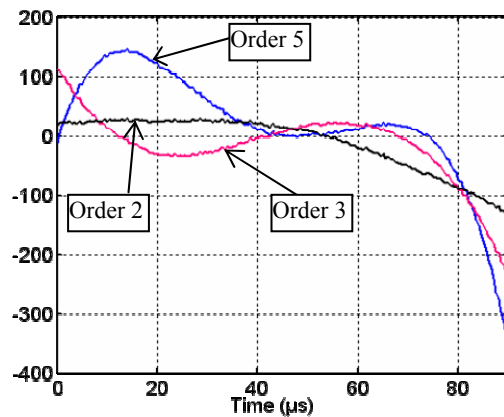


Fig. 6. Time variations of the current approximation errors.

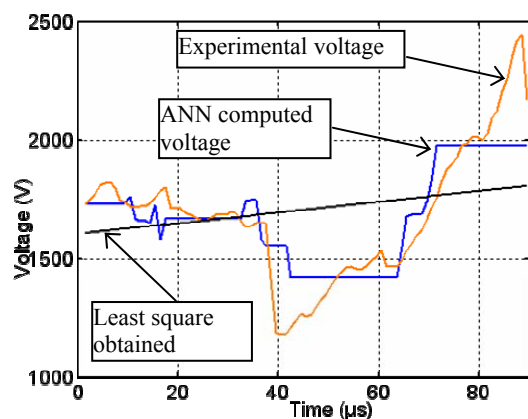


Fig. 7. Arc voltage approximation by a polynomial of 1<sup>st</sup> order.

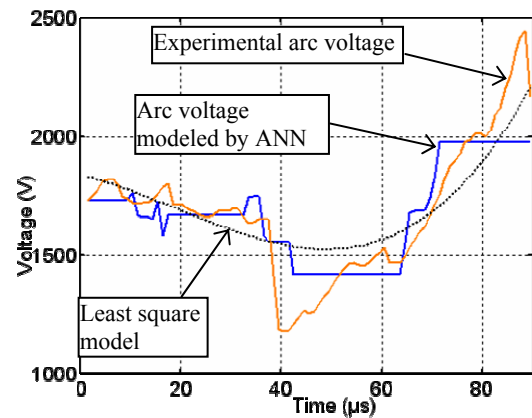


Fig. 8. Arc voltage approximated by a polynomial of 5<sup>th</sup> order.

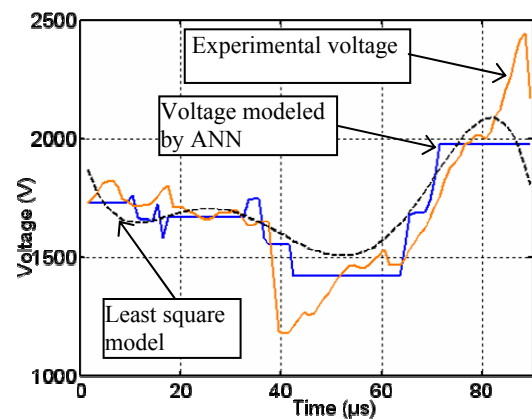


Fig. 9. Arc voltage approximation by a polynomial of order 10.

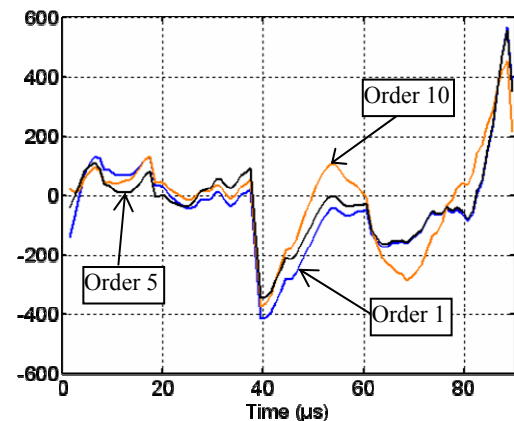


Fig. 10. Arc voltage approximation error evolvement.

## 5. Conclusion

The association of least square techniques with artificial neural networks appears well adapted to modeling of arc quenching in high voltage circuit breakers. One notes that the current models are better adapted with a linear regression

curve, whereas the arc voltage is best modeled by elevated degree polynomial functions. However, we can add that the retained neural model remains limited for electric arc extinction modeling if we consider the important recorded gaps with regard to the experimental data.

One of future challenges of this work is to take in account other physical parameters, in addition to the breaking duration.

## References

- [1] Chen Xiaoning, Contrôle optimal d'un disjoncteur de puissance. Visualisation ; mise en œuvre d'un réseau de neurones, doctorat thesis, Université de Cergy-Pontoise, France, 2000.
- [2] S. Vacquié, Arc électrique, Techniques de l'Ingénieur, Traité de Génie Electrique D 2870, 1995.
- [3] CIGRE Working Group 13-01, Applications of Black Box Modelling to Circuit Breaker, *Electra*, 149, 41-71, 1993
- [4] L. Reynard, Modélisation tridimensionnelle de l'arc électrique dans un disjoncteur basse tension, doctorat thesis, Ecole centrale de Lyon, France, 2006.
- [5] A. Ziani and H. Moulai, "Extinction properties of electric arcs in high voltage circuit breakers", *Journal of Physics D: Applied Physics*, Vol. 42, No 10, 2009.
- [6] A. Ziani and H. Moulai, '0D Model of thermal exchanges at the opening of an SF6 high voltage circuit breaker'', ACTA Press, Proc. of the Int. Conf. on Power and Energy systems, IASTED Europe's 2009, 07-09 September 2009, Palma de Mallorca, Spain.
- [7] E. Davalo and P. Naim, *Les Réseaux de Neurones*, Ed. Eyrolles, 1993
- [8] S. Chen and S.A Billings, "Neural networks for nonlinear system modelling and identification", *International Journal of Control*, Vol. 56, pp.319-346, 1992.
- [9] K. Hornick, M. Stinchcombe, and White, "Multilayer feedforward networks are universal approximators", *Neural Networks*, Vol. 2, pp.359-366, 1989.
- [10] J.Parks, and I.W. Sandberg, "Universal approximation using radial-basis function networks", *Neural computation*, Vol 3, pp.246-257, 1991.
- [11] P. H. Schavemaker and L. Van der Sluis, "An improved Mayr type arc model based on current zero measurement", *IEEE Trans. Power Delivery*, 15, 580-584, 2000.
- [12] J. L. Guardado and S. G. Maximov, "An Improved Arc Model Before Current Zero Based on the Combined Mayr and Cassie Arc Models", *IEEE Trans. Power Delivery*, Vol. 20, N°1, January 2005.
- [13] B.Krose and P. Van der Smagt, *An introduction to neural network*, ebook of The University of Amsterdam, November 1996.
- [14] A. Ziani and H. Moulai, "Application of artificial neural networks for electric arc extinction modeling in high voltage circuit breakers", MELECON 2010, 15th IEEE Mediterranean Electrotechnical Conference, 26-28-April 2010, Valetta, Malta, pp 248-252,.