FERROMAGNETIC HYSTERESIS MODELLING WITH ADAPTIVE NEURO–FUZZY INFERENCE SYSTEM

Mourad Mordjaoui * — **Mabrouk Chabane** *** — **Bouzid Boudjema** **

Hysteresis modeling is an important criterion in defining the electromagnetic properties of magnetic materials. Many models are available to investigate those characteristics but they tend to be complex and difficult to implement. In this paper a new qualitative hysteresis model for the ferromagnetic core is presented, based on the function approximation capabilities of an adaptive neuro fuzzy inference system (ANFIS). The proposed ANFIS model combined the neural network adaptive capabilities and the fuzzy logic qualitative approach can restore the hysteresis curve with a little RMS error. Therefore, the accuracy of the model developed is good as the quality of the data used.

K e y w o r d s: ANFIS modeling technique, magnetic hysteresis, Jiles-Atherton model, ferromagnetic core

1 INTRODUCTION

Computation of electrical machines requires a deep knowledge of material characteristics used in their construction. Many researchers were interested in the area of magnetic hysteresis modeling. The Preisach model was the first mathematical hysteresis model developed to describe the relationship between the magnetization M and the magnetic field H [1], then Jiles and Atherton presented a physical model [2]. Artificial intelligence has also been applied to the modeling of magnetic hysteresis and parameters identification of these models such as neural network and genetic algorithm [3–13]. Like neural networks, fuzzy logic can be conveniently used to approximate any arbitrary functions [14–16]. Neural networks can learn from data but knowledge learned can be difficult to understand. Models based on fuzzy logic are easy to understand but they do not have learning algorithms; learning has to be adapted from other technologies. A Neuro-Fuzzy model can be defined as a model built using a combination of fuzzy logic and neural networks. Recently, there has been a remarkable advance in the development of Neuro-Fuzzy models, as it is described in [17–19]. One of the most popular and well documented Neuro-Fuzzy systems is ANFIS, which has a good software support [20]. Jang [21–23] present the ANFIS architecture and application examples in modeling a nonlinear function, dynamic system identification and chaotic time series prediction. Given its potential in building fuzzy models with good prediction capabilities, the ANFIS architecture was chosen for modeling magnetic hysteresis in this work. In this paper, the Jiles-Atherton hysteresis model was introduced and a neuro-fuzzy system was proposed to modeling the ferromagnetic material hysteretic behavior. The proposed approach will be presented in the following sections.

2 JILES–ATHERTON HYSTERESIS MODEL

The Jiles-Atherton model is a physically based model that includes the different mechanisms that take place at magnetization of a ferromagnetic material. The magnetization M is represented as the sum of the irreversible magnetization M_{irr} due to domain wall displacement and the reversible magnetization M_{rev} due to domain wall bending [2]. The anhysteretic magnetization M_{an} in (2) follows the Langevin function [3], which is a nonlinear function of the effective field:

$$H_e = H + aM\,, \tag{1}$$

$$M_{an} = M_s\Big(\coth\Big(\frac{H_e}{a}\Big) - \frac{a}{H_e}\Big). \tag{2}$$

The rate of change of the reversible component is proportional to the rate of the difference between the hysteretic component and the total magnetization [4]. Combining the irreversible and reversible components of magnetization, the differential equation for the rate of change of the total magnetization is given by:

$$\frac{\mathrm{d}M}{\mathrm{d}H} = \frac{1}{1+c}\,\frac{M_{an}-M}{\frac{k\delta}{\mu_0} - \alpha(M_{an}-M)} + \frac{c}{c+1}\,\frac{\mathrm{d}M_{an}}{\mathrm{d}H}\,. \tag{3}$$

Before using the J-A model, five parameters must be determined:

- α: a mean field parameter defining the magnetic coupling between domains and all types of magnetic anisotropy in the material, and is required to calculate the effective magnetic field, H_e (1) composed by the applied external field and the internal magnetization.
- M_s: saturation magnetization.
- a: Langevin parameter.

* Electrical Engineering Department, University of Skikda, Algeria, ** Physics Department, University of Skikda, Algeria *** Electrical Engineering Department, University of Batna, Algeria; Mordjaoui_mourad@yahoo.fr, Boudjema_b@yahoo.fr, Machabane@yahoo.com

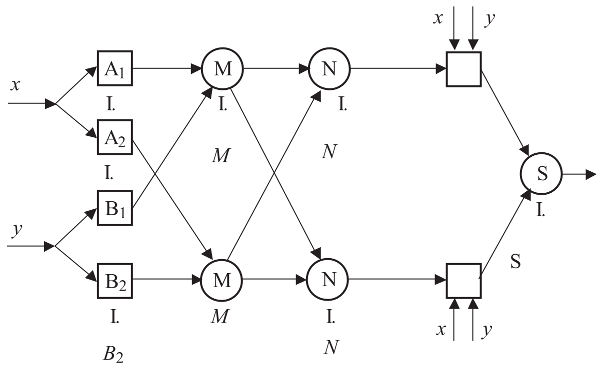

Fig. 1. ANFIS architecture

These two parameters defined a Langevin function needed in the equation describing anhysteretic curve.

- k: parameter defining the pinning site density of domain walls. It is assumed to be the major contribution to hysteresis.
- c: parameter defining the amount of reversible magnetization due to wall bowing and reversal rotation, included in the magnetization process.

δ is a directional parameter and takes $+1$ for increasing field ($\mathrm{d}H/\mathrm{d}t > 0$) and -1 for decreasing field ($\mathrm{d}H/\mathrm{d}t < 0$).

3 ADAPTIVE NEURO–FUZZY INFERENCE SYSTEM (ANFIS)

An adaptive Neuro-Fuzzy inference system is a cross between an artificial neural network and a fuzzy inference system. An artificial neural network is designed to mimic the characteristics of the human brain and consists of a collection of artificial neurons. An adaptive network is a multi-layer feed-forward network in which each node (neuron) performs a particular function on incoming signals. The form of the node functions may vary from node to node. In an adaptive network, there are two types of nodes: adaptive and fixed. The function and the grouping of the neurons are dependent on the overall function of the network. Based on the ability of an ANFIS to learn from training data, it is possible to create an ANFIS structure from an extremely limited mathematical representation of the system.

3.1 Architecture of ANFIS

ANFIS is a fuzzy Sugeno model put in the framework of adaptive systems to facilitate learning and adaptation [18]. Such framework makes the ANFIS modeling more systematic and less reliant on expert knowledge. To present the ANFIS architecture, we suppose that there are two input linguistic variables (x, y) and each variable has two fuzzy sets (A_1, A_2) and (B_1, B_2) as is indicated in Fig. 1, in which a circle indicates a fixed node, whereas a square indicates an adaptive node.

Then a Takagi-Sugeno-type fuzzy *if-then* rule could be set up as:

Rule i: If (x is A_i) and (y is B_I) then ($f_i = p_i x + q_i y + r_i$)

f_i are the outputs within the fuzzy region specified by the fuzzy rule. p_i, q_i and r_i are the design parameters that are determined during the training process.

Some layers of ANFIS have the same number of nodes, and nodes in the same layer have similar functions. Output of nodes in layer-l is denoted as O_i^1, where 1 is the layer number and i is the neuron number of the next layer. The function of each layer is described as follows:

• **Layer 1**: In this layer, all the nodes are adaptive nodes. The outputs of layer 1 are the fuzzy membership grade of the inputs, which are given by:

$$O_i^1 = \mu_{A_i}(x)\,, \qquad i = 1, 2\,, \tag{4}$$

$$O_i^1 = \mu_{B_{i-2}}(y)\,, \qquad i = 3, 4\,, \tag{5}$$

where $\mu_{A_i}(x)$, $\mu_{B_{i-2}}(y)$ can adopt any fuzzy membership function. For example, if the bell shaped membership function is employed, $\mu_{A_i}(x)$ is given by:

$$\mu_{A_i}(x) = \frac{1}{1 + \left\{\left(\frac{x - c_i}{a_i}\right)^2\right\}^{b_i}} \tag{6}$$

where a_i, b_i and c_i are the parameters of the membership function, governing the bell shaped functions accordingly.

• **Layer 2**: Each node computes the firing strengths of the associated rules. The output of nodes in this layer can be presented as:

$$O_i^2 = \omega_i = \mu_{A_i}(x)\mu_{B_i}(y)\,, \quad i = 1, 2\,. \tag{7}$$

• **Layer 3**: In this third layer, the nodes are also fixed nodes. They play a normalization role to the firing strengths from the previous layer. The outputs of this layer can be represented as:

$$O_i^3 = \overline{\omega}_i = \frac{\omega_i}{\omega_1 + \omega_2}\,, \quad i = 1, 2 \tag{8}$$

which are the so-called normalized firing levels.

• **Layer 4**: The output of each adaptive node in this layer is simply the product of the normalized firing level and a first order polynomial (for a first order Sugeno model). Thus, the outputs of this layer are given by:

$$O_i^4 = \overline{\omega}_i f_i = \overline{\omega}_i \big(p_i x + q_i y + r_i\big)\,, \quad i = 1, 2\,. \tag{9}$$

• **Layer 5**: Finally, layer five, consisting of circle node labeled with S. is the summation of all incoming signals. Hence , the overall output of the model is given by:

$$O_i^5 = \sum_{i=1}^{2} \overline{\omega}_i f_i = \frac{\sum_{i=1}^{2} \omega_i f_i}{\omega_1 + \omega_2}\,. \tag{10}$$

From the architecture of ANFIS, we can observe that there are two adaptive layers the first and the fourth. In the first layer, there are three modifiable parameters $\{a_i, b_i, c_I\}$, which are related to the input membership functions. These parameters are the so-called premise parameters. In the fourth layer, there are also three modifiable parameters $\{p_i, q_i, r_I\}$, pertaining to the first order polynomial. These parameters are so-called consequent parameters [21, 22].

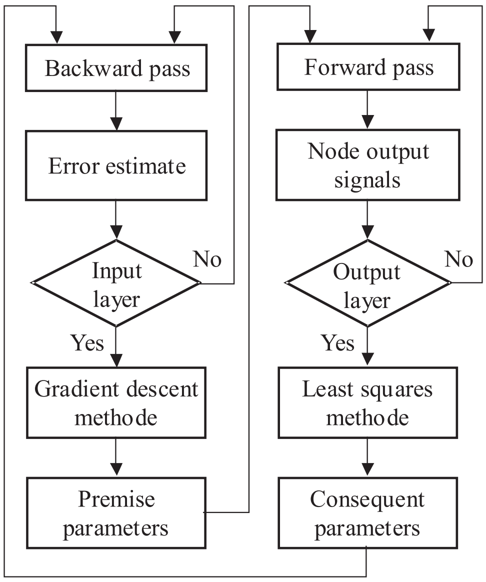

Fig. 2. ANFIS training algorithm for adjusting production rules parameters

3.2 Learning algorithm of ANFIS

The task of training algorithm for this architecture is tuning all the modifiable parameters to make the ANFIS output match the training data. Note here that a_i, b_i and c_i describe the sigma, slope and the center of the bell MF's, respectively. If these parameters are fixed, the output of the network becomes:

$$f = \frac{\omega_1}{\omega_1 + \omega_2} f_1 + \frac{\omega_1}{\omega_1 + \omega_2} f_2 \,. \tag{11}$$

Substituting Eq. (8) into Eq. (11) yields:

$$f = \overline{\omega}_1 f_1 + \overline{\omega}_2 f_2 \,. \tag{12}$$

Substituting the fuzzy if-then rules into equation (12), it becomes:

$$f = \overline{\omega}_1 (p_1 x + q_1 y + r_1) + \overline{\omega}_2 (p_2 x + q_2 y + r_2) \,. \tag{13}$$

After rearrangement, the output can be expressed as:

$$f = (\overline{\omega}_1 x) p_1 + (\overline{\omega}_1 y) q_1 + (\overline{\omega}_1) r_1 + (\overline{\omega}_2 x) p_2 + (\overline{\omega}_2 y) q_2 + (\overline{\omega}_2) r_2 \,. \tag{14}$$

This is a linear combination of the modifiable parameters. For this observation, we can divide the parameter set S into two sets:

$S = S_1 \oplus S_2$,

$S =$ set of total parameters,

$S_1 =$ set of premise (nonlinear) parameters,

$S_2 =$ set of consequent (linear) parameters,

$\oplus$: Direct sum.

For the forward path (see Fig. 1), we can apply least square method to identify the consequent parameters. Now for a given set of values of S_1, we can plug training data and obtain a matrix equation:

$$A\Theta = y \tag{15}$$

where Θ contains the unknown parameters in S_2. This is a linear square problem, and the solution for Θ, which is minimizes $\|A\Theta = y\|$, is the least square estimator:

$$\Theta^* = \left(A^\top A\right)^{-1} A^\top y \tag{16}$$

we can use also recursive least square estimator in case of on-line training. For the backward path (see Fig. 1), the error signals propagate backward. The premise parameters are updated by descent method, through minimizing the overall quadratic cost function

$$J(\Theta) = \frac{1}{2} \sum_{k=1}^{N} \left[y(k) - \hat{y}(k, \Theta)\right]^2 \tag{17}$$

in a recursive manner with respect $\Theta_{(S2)}$. The update of the parameters in the i^{th} node in L^{th} layer can be written as:

$$\widehat{\theta}_i(k) = \widehat{\theta}_i^L(k-1) + \frac{\partial E(k)}{\partial \widehat{\Theta}_i^L(k)} \tag{18}$$

where η is the learning rate and the gradient vector

$$\frac{\partial^+ E}{\partial \widehat{\Theta}_i^L} = \varepsilon_{L,i} \frac{\partial \hat{z}_{L,i}}{\partial \widehat{\Theta}_i^L} \,, \tag{19}$$

$\partial \hat{z}_{L,i}$ being the node's output and $\varepsilon_{L,i}$ is the backpropagated error signal. Figure 2 presents the ANFIS training algorithm for adjusting production rules parameters.

4 APPROXIMATING MAGNETIC HYSTERESIS

4.1 Simulation

The differential equation (3), which in its original form has derivatives with respect to H, was reformulated into a differential equation in time by multiplying the left and the right sides by $\mathrm{d}H/\mathrm{d}t$, thus resulting in:

$$\frac{\mathrm{d}M}{\mathrm{d}t} = \frac{1}{1+c} \frac{\mathrm{d}H}{\mathrm{d}t} \frac{M_{an} - M}{\frac{\delta k}{\mu_0} - \alpha\left(M_{an} - M\right)} + \frac{c}{1+c} \frac{\mathrm{d}M_{an}}{\mathrm{d}t} \,. \tag{20}$$

This reformulation allows for the determination of mag-

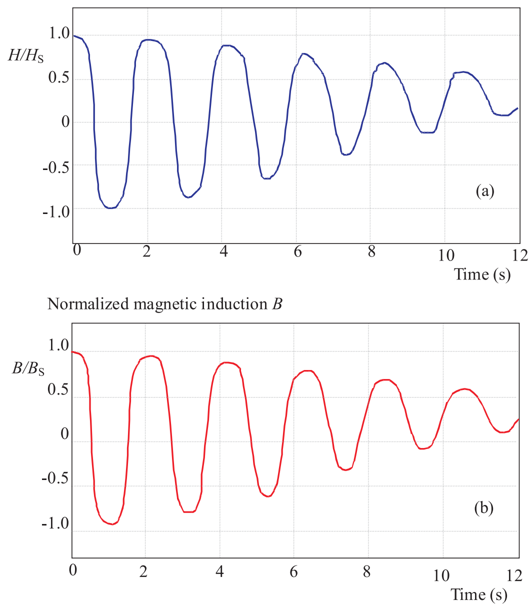

Fig. 3. (a) – Normalized magnetic field versus time (b) – Normalized magnetic induction versus time

netization by use of Runge Kutta method in Matlab environment. To calculate the magnetic flux density B from M and H, the following constitutive law of the magnetic material property is used.

$$B = \mu H = \mu_0 \mu_r H = \mu_0 (H + M) . \tag{21}$$

Where $\mu_0 = 4\pi \times 10^{-7}$ (H/m) is the permeability of free space and μ_r is the relative permeability.

The $B(H)$ curve result of simulation of the Jiles-Atherton model will be used as 'experimental data' to be approximate by proposed Neuro-Fuzzy model.

4.2 Proposed model

In this section, the learning ability of ANFIS is verified by approximating a hysteresis of magnetic material. The data set used as input/output pairs for Anfis were generated by Jiles Atherton model for ferrite core described in [24] with sinusoidal magnetic field as an input $H(t)$ and magnetic field $B(t)$ as shown in Fig. 3a and Fig. 3b.

Our purpose is to predict the magnetic hysteresis cycles using 12 candidate inputs to ANFIS : $B(t-i)$ for $i = 1:5$, and $H(t-j)$ for $j = 1:7$. Converted from the original data sets containing 422 $[H(t)\ B(t)]$ pairs.

In the first time, we suppose that there are two inputs for ANFIS and we have to construct 35 ANFIS models (5×7) with various input combinations, and then select the one with the smallest training error for further parameter-level fine tuning. In Table 1 we can see that the ANFIS with $B4$ and $H1$ (in red) as inputs has the smallest training error, so it is reasonable to choose this ANFIS for further parameter tuning. Note that each ANFIS has four rules, and the training took only one epoch each to identify linear parameters. Let us note that the computing time for selecting the good model is 5.03882 s.

Table 1. Training and checking error for all models.

Model	Training error	Checking error
B1 H1	0.00003362751798	0.00004197186908
B1 H2	0.00959687404061	0.01093765284221
B1 H3	0.01656339121598	0.01464519925359
B1 H4	0.02303817217175	0.01875072447019
B1 H5	0.02955482263776	0.02737874889745
B1 H6	0.03592909843370	0.04070823636094
B1 H7	0.04234687502187	0.06046603382639
B2 H1	0.00003454986338	0.00004235702777
B2 H2	0.01326113905508	0.02596375295441
B2 H3	0.01816578945136	0.01960397432510
B2 H4	0.02421933820921	0.02017463297709
B2 H5	0.03048223398339	0.02534746376602
B2 H6	0.03631870393183	0.03506607375679
B2 H7	0.04206553808394	0.05091660789003
B3 H1	0.00002714382722	0.00002969609241
B3 H2	0.01125282874177	0.02760786485845
B3 H3	0.02519247143986	0.05013635694792
B3 H4	0.02625830973011	0.02661118233221
B3 H5	0.03217201528397	0.02597623870064
B3 H6	0.03760746620563	0.03182176441713
B3 H7	0.04262648998736	0.04393612682667
B4 H1	0.00002151667942	0.00002941797497
B4 H2	0.00946786265086	0.01284544986247
B4 H3	0.02265729876255	0.05706885484696
B4 H4	0.03575210537745	0.07229359571869
B4 H5	0.03458484870981	0.03288908255780
B4 H6	0.03983270473694	0.03162888402750
B4 H7	0.04433645678973	0.03955927980351
B5 H1	0.00002820696869	0.00003255142152
B5 H2	0.00868269284842	0.00825133809164
B5 H3	0.01579989242722	0.01323395619449
B5 H4	0.02249300689915	0.01945532082103
B5 H5	0.02931964087956	0.03153342378357
B5 H6	0.03616365495642	0.04883984122394
B5 H7	0.04295327484075	0.07317450661150

After selection of the good and adapted model, we made train the network 100 epochs, for this purpose we have used 211 pairs as training data and 211 pairs for checking, shown in Fig. 3.

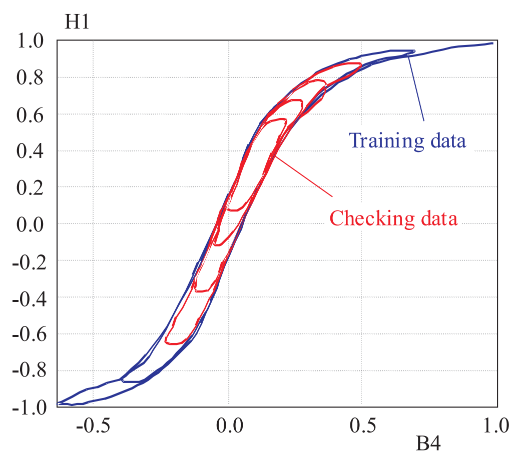

Fig. 4. Data distribution

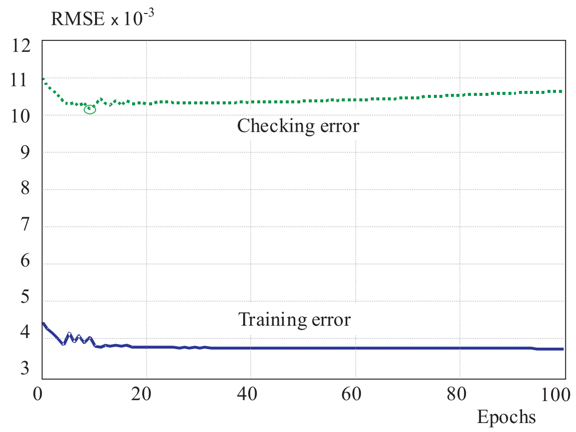

Fig. 5. Error curves

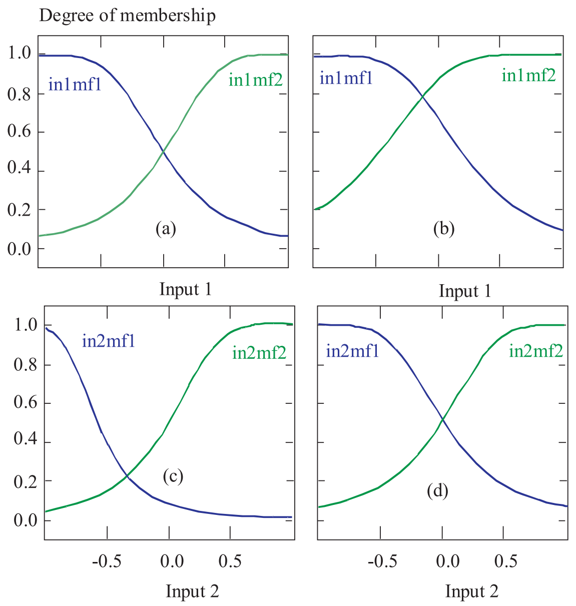

Fig. 6. Initial and final generalized bell-shaped membership function of input 1 and 2 for the Best model. (a)– Initial, and (b)– final MFs on "x", (c)– Initial, and (d) – final MFs on "y"

The number of MFs assigned to each input of the ANFIS was set to two bell type, so the number of rules is 04. The training was run for 100 iterations, the network performance were evaluated on the checking set after every iteration, by calculating the root-mean-square errors (RMSE):

$$RMSE = \sqrt{\frac{\sum_{k=1}^{K}(y_k - \hat{y}_k)}{K}} \tag{22}$$

where k is the pattern number, $k = 1, \dots, K$. The RMSE was also evaluated on training data set in every iteration. The optimal number of iteration was obtained when checking RMSE has reached its minimum value 0.01019 after 10 epochs, see Fig. 5.

Figure 6 depicts the initial and final membership functions for each input variable. The anfis used here contains a total of 24. fitting parameters, of which 12 are premise (nonlinear) parameters and 12 are consequent (linear) parameters. Table 2 summarizes all characteristics of the network used.

Table 2. ANFIS caracteristics

Number of nodes	21
Number of linear parameters	12
Number of nonlinear parameters	12
Total number of parameters	24
Number of training data pairs	211
Number of checking data pairs	211
Number of fuzzy rules	04

The ANFIS shown in Fig. 1 was implemented by using MATLAB software package (MATLAB version 6.5 with fuzzy logic toolbox), it uses 422 training data in 100 training periods and the step size for parameter adaptation had an initial value of 0.1. The steps of parameter adaptation and Anfis surface are shown in Fig. 7 and Fig. 8 respectively.

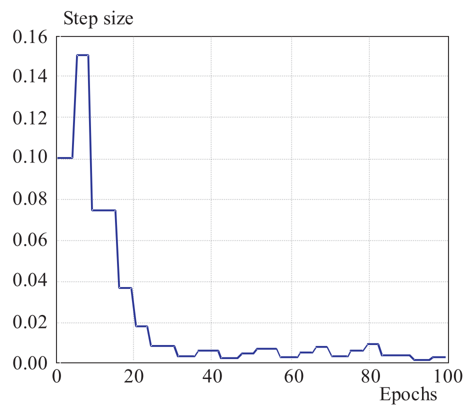

Fig. 7. Adaptation of parameter steps of ANFIS

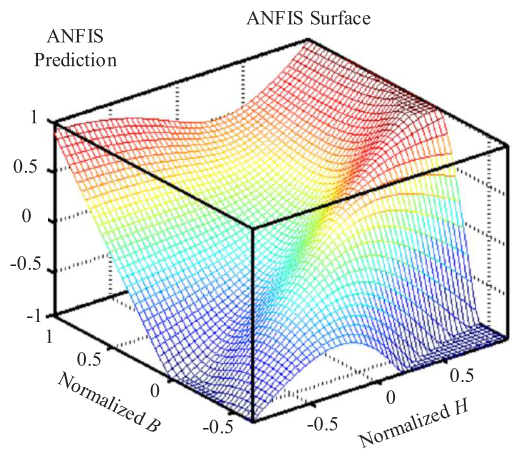

Fig. 8. ANFIS surface

The obtained ANFIS network was evaluated on, the complete data set using $Ts = 0.362$ s and resulted in a good prediction (Fig. 9) with $RMSE = 0.003834$.

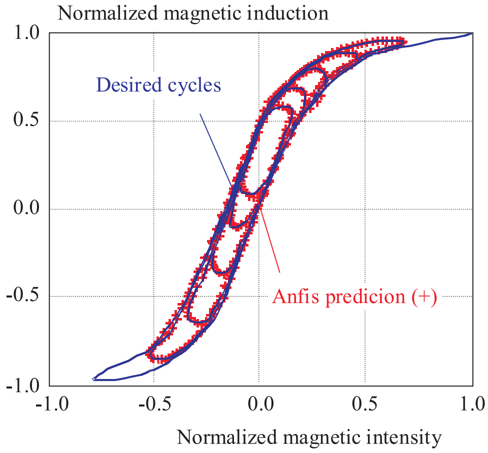

Fig. 9. ANFIS surface

5 CONCLUSION

The neuro-fuzzy model developed for predicting static hysteresis of ferromagnetic material provides a means for magnetic hysteretic behavior knowledge. Results revealed that the proposed model is more closely to Jiles-Atherton model. The collection of well-distributed, sufficient, and accurately measured input data is the basic requirement to obtain an accurate model. The adequate functioning of ANFIS depends on the sizes of the training set and test set. Future work will be focused in implementing this model into electromagnetic devices analysis by finite element method.

References

[1] PREISACH, F.: Über die magnetische Nachwirkung, Zeitschrift für physik **94** No. 56 (1935), 277. (in German)

[2] JILES, D. C.—ATHERTON, D. L.: Theory of Ferromagnetic Hysteresis, J. Magn. Magn. Mater. **61** (1986), 48-60.

[3] WILSON, K. P. R.—ROSS, J. N.—BROWN, A. D.: Optimizing the JilesAtherton Model of Hysteresis by a Genetic Algorithm, IEEE Trans. Magn. **37** (2001), 989-993.

[4] SALVINI, A.—FULGINEI, F. RIGANTI: Genetic Algorithms and Neural Networks Loops, IEEE Trans. Magn. **38** (2002), 873-876.

[5] SALVINI, A.—COLTELLI, C.: Prediction of Dynamic Hysteresis under Highly Distorted Exciting Fields by Neural Networks and Actual Frequency Transplantation, IEEE Trans. Magn. **37** (2001), 3315-3319.

[6] Del VECCHIO, P.—SALVINI, A.: Neural Networks and Fourier Descriptor Macromodeling Dynamic Hysteresis, IEEE Trans. Magn. **36** (2000), 1246-1249.

[7] XU, Q.—REFSUM, A.: Neural Network for Represention of Hysteresis Loops, IEE proc-Sci Meas Technol **144** No. 6 (1997), 263–266.

[8] SALIAH, H. H.—LOWTHER, D. D.: A Neural Network Nodel of Magnetic Hysteresis for Computational Magnetics, IEEE Trans. Magn. **33** No. 05 (1997), 4146–4148.

[9] MAKAVEEV, D.—DUPRE, L.—De WULF, M.—MELKEBEEK, J.: Modeling of Quasistatic Hysteresis with Feed-Forwad Neural Networks, Journal of Applied Physics **99** No. 11 (2001), 6737–6739.

[10] MAKAVEEV, D.—DUPRE, L.—De WULF, M.—MELKEBEEK, J.: Combined PreisachMayergoyz-Neural-Network Vector Hysteresis Model for Electrical Steel Sheets, Journal of Applied Physics **93** No. 10 (2003), 6738–6740.

[11] SAGHAFIFAR, M.—NAFALSKI, A.: Magnetic Hysteresis Modeling using Dynamic Neural Networks, Japan Society of Applied Electromagnetics and Mechanics (JSAEM) Studies in Applied Electromagnetics and Mechanics, vol. 14, JSAEM, Kanazawa, Japan, 2002, pp. 293–299.

[12] KUCZMANN, M.—IVÁNYI, A.: A New Neural-Network-Based Scalar Hysteresis Model, IEEE Trans. on Magn. **38** No. 2 (2002), 857–860.

[13] KUCZMANN, M.—IVÁNYI, A.: Neural Network Model of Magnetic Hysteresis, Compel **21** No. 3 (2002), 364–376.

[14] ZADEH, L. A.: Fuzzy Sets, Information and Control **8** (1965), 338-353.

[15] BUCKLEY, J. J.—HAYASHI, Y.: Hybrid Fuzzy Neural Nets are Universal Approximators, Proc. IEEE Int. Conf. on Fuzzy Systems 1994, Orlando, FL, 1994, pp. 238-243.

[16] KOSKO, B.: Fuzzy Systems as Universal Approximators, Proc. IEEE Int. Conf. on Fuzzy Systems, San Diego, CA, 1992, pp. 1153-1162.

[17] YEN, J.—LANGARI, R.: Fuzzy Logic. Intelligence, Control and Information, Prentice Hall, 1999.

[18] JANG, J. S.—SUN, C.-T.—MIZUTANI, E.: Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997.

[19] ABRAHAM, A.: Neuro-Fuzzy Systems: State-of-the-art Modeling Techniques, In: Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence (J. Mira and A. Prieto, eds.), Springer-Verlag, 2001, pp. 269–276.

[20] The MathWorks, Inc., Fuzzy Logic Toolbox. The MathWorks, Inc., 1998.

[21] JANG, J. S.: ANFIS: Adaptive-Network-Based Fuzzy Inference Systems, IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics **23** (1993), 665–685.

[22] JANG, J.-SR.: Self-Learning Fuzzy Controllers Based on Temporal Backpropagation, IEEE Trans. Neural Netw. **3** No. 5 (1992), 714-723.

[23] ROGER JANG, J. S.—SUN, C. T.: Neuro-Fuzzy Modeling And Control, Proceeding of the IEEE **83** No. 3 (1995).

[24] Del MORAL HERNANDEZ, E.—MURANAKA, C. S.—CARDOSO, J. R.: Identification of the Jiles-Atherton Model Parameters Using Random and Deterministic Searches, Physica **B** **275** (2000), 212–215.

Received 7 April 2007

Mourad Mordjaoui was born in Skikda, Algeria, in 1968. He received the degree of MSc in electrical engineering, from university of Skikda in 1998 and the PhD degree in electrical engineering from University of Batna, Algeria, 2008. Currently, he is a lecturer in Electrical Science and he is also a member of the research laboratory (LRPCSI) at the university of Skikda Algeria. He is principally concerned with research into modeling and simulation of frequency effects in ferromagnetic material using soft computing techniques.

Mabrouk Chabane and **Bouzid Boudjema**, biographies not supplied.