World Academy of Science, Engineering and Technology
International Journal of Electrical and Computer Engineering
Vol:2, No:2, 2008

Inverse Problem Methodology for the Measurement of the Electromagnetic Parameters Using MLP Neural Network

T. Hacib, M. R. Mekideche and N. Ferkha

***Abstract*—**This paper presents an approach which is based on the use of supervised feed forward neural network, namely multilayer perceptron (MLP) neural network and finite element method (FEM) to solve the inverse problem of parameters identification. The approach is used to identify unknown parameters of ferromagnetic materials. The methodology used in this study consists in the simulation of a large number of parameters in a material under test, using the finite element method (FEM). Both variations in relative magnetic permeability and electrical conductivity of the material under test are considered. Then, the obtained results are used to generate a set of vectors for the training of MLP neural network. Finally, the obtained neural network is used to evaluate a group of new materials, simulated by the FEM, but not belonging to the original dataset. Noisy data, added to the probe measurements is used to enhance the robustness of the method. The reached results demonstrate the efficiency of the proposed approach, and encourage future works on this subject.

***Keywords*—**Inverse problem, MLP neural network, parameters identification, FEM.

I. Introduction

INVERSE problems in electromagnetic are usually formulated and solved as optimization problems, so iterative methods are commonly used approaches to solve this kind of problems [1]. These methods involve solving well behaved forward problem in a feedback loop. The numerical models such as FEM are used to represent the forward process. However, iterative methods using the numerical based forward models are computationally expensive. Recently, artificial neural networks (ANNs) are introduced to solve the inverse problems in most of the research applications in industrial nondestructive testing, mathematical modeling, medical diagnostics and detection of earthquakes [2-6].

Electromagnetic inverse problems can sometimes be stated as simply as the following: if there is an electromagnetic device, it is easy to calculate the magnetic induction in any region of the device. What, about taking some values of magnetic induction to predict physical parameters in a region of the electromagnetic device. Since, the inverse problem is highly nonlinear and without formulations to follow, it is very difficult to construct an effective inversion algorithm. An ANN, however, has the following properties: nonlinearity, input-output mapping, fault tolerance and most important, learning from examples.

ANNs consist of a large number of simple processing elements called neurons or nodes. Each neuron is connected to other neurons by means of directed links, each with an associated weight [7]. The weights represent information being used by the network to solve a problem. The ANN essentially determines the relationship between input and output by looking at examples of many input-output pairs. In learning processes, the actual output of the ANN is compared to the desired output. Changes are made by modifying the connection weights of the network to produce a closer match. The procedure iterates until the error is small enough [8].

In this paper we present a new method for the robust estimation of electromagnetic parameters. The method is based on the use of FEM and ANN scheme. The network is trained by a large number of parameters in a metallic wall simulated using the FEM. The obtained results are then used to generate the training vectors for ANN. The trained network is used to identify new electromagnetic parameters in the metallic wall, which not belong to the original dataset. The network weights can be embedded in an electronic device, and used to identify parameters in real pieces, with similar characteristics to those of the simulated ones.

For the methodology presented here, the measured values are independent of the relative motion between the probe and the material under test. In other words, the movement is necessary only to change the position of the probes, to acquire the field's values, which are necessary for the identification of new parameters. The kinds of parameter we have investigated are relative magnetic permeability and electrical conductivity of the material under test. For the purpose of the paper, the data set was generated considering 20 variations in the relative magnetic permeability and 15 variations in the electrical conductivity, performing at least 300 finite elements simulations.

Manuscript received July 05, 2007.

T. Hacib, M. R. Mekideche and N. Ferkha are with Electrical Engineering Department, Electrical engineering study and modelling laboratory, Jijel University, 18000 Algeria (phone: +213-34 50 11 89; fax: +213-34 50 26 89; e-mail: tarik_hacib@mail.univ-jijel.dz, mek_moh@mail.univ-jijel.dz and nassira_ferkha@yahoo.fr).

II. Neural Network Architecture

ANNs are parallel distributed information processing models that can recognize highly complex patterns within available data. An ANN is an information processing system that has certain performance characteristics in common with biological neural networks and therefore, each network is a collection of neurons that are arranged in specific formations. The basic elements of neural network comprise neurons and their connection strengths (weights). One of the attractive features of ANNs is their capability to adapt themselves to special environmental conditions by changing their connection strengths or structure. Years of studies have shown that ANNs exhibit a surprising number of the brain's characteristics. For example, they learn from experience, generalize from previous examples, and abstract essential characteristics from inputs containing irrelevant data. In this paper we choose the back-propagation method to demonstrate the potential of ANNs to solve electromagnetic inverse problems of parameters identifications [9].

One of the most influential developments in ANN was the invention of the back-propagation algorithm, which is a systematic method for training multilayer ANNs [10]. The standard back-propagation learning algorithm for feed-forward networks aims to minimize the mean squared error defined over a set of training data. In feed-forward ANNs neurons are arranged in a feed-forward manner, so each neuron may receive an input from the external environment or from the neurons in the former layer, but no feedback is formed. The network architecture for a feed forward network consists of layers of processing nodes. The network always has an input layer, an output layer and at least one hidden layer. There is no theoretical limit on the number of hidden layers but typically there will be one or two. In our case, there is only one hidden layer. Every neuron in each layer of the network is connected to every neuron in the adjacent forward layer. A neuron's activity is modeled as a function of the sum of its weighted inputs, where the function is called the activation function, which is typically nonlinear, thus giving the network nonlinear decision capability. Each layer is fully connected to the succeeding layer. The arrows indicate flow of information (Fig.1) [11, 12].

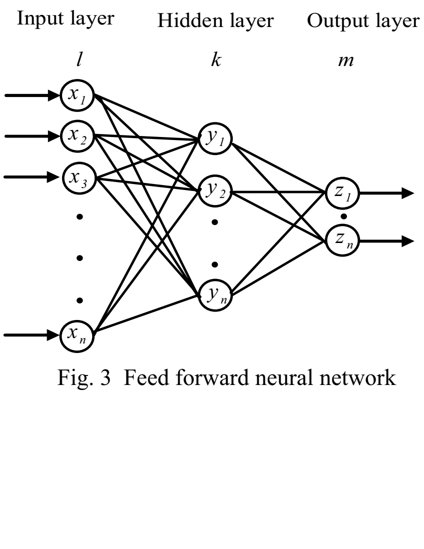

Fig. 3 Feed forward neural network

where n_l is the number of neurons in the input layer, n_H is the number of neurons in the hidden layer, n_O is the number of neurons in the output layer, x_l are the inputs to the input layer where $l = 1,\ldots,n_l$, y_k is the value of the hidden layer where $k = 1,\ldots,n_H$, z_m is the value of the output layer where $m = 1,\ldots,n_O$. $w_{lk}^{[1]}$ is the weight connecting the $l\,th$ neuron in the input layer to $k\,th$ neuron in the hidden layer, and $w_{km}^{[2]}$ is the weight connecting the $k\,th$ neuron in the hidden layer to the $m\,th$ neuron in the output layer. The nodes of the hidden and output layer are:

$$y_k = f\left(\sum_{l=1}^{n_l} w_{lk}\, x_l\right) \qquad k = 1,\ldots,n_H \tag{1}$$

and

$$z_m = f\left(\sum_{k=1}^{n_k} w_{mk}\, y_k\right) \qquad m = 1,\ldots,n_O \tag{2}$$

where the activation function f is traditionally the Sigmoid function but can be any differentiable function. The Sigmoid function is defined as

$$f(x) = \frac{1}{(1+e^{-x})} \tag{3}$$

This activation function is depicted in Fig. 2.

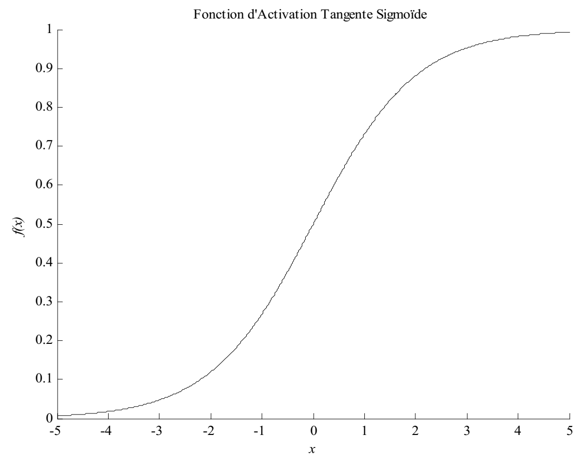

Fig. 6 Sigmoid activation function

The back-propagation method is based on finding the outputs at the last (output) layer of the network and calculating the errors or differences between the desired outputs and the current outputs. When the outputs are different from the desired outputs, corrections are made in the weights, in proportion to the error.

$$\Delta w_{km}^{[2]} = y_k f^{'}(z_m)(z_m - d_m) \tag{4}$$

where d_m represent the desired output, $k = 1,\ldots,n_H$, $m = 1,\ldots,n_O$ and

$$f^{'}(x)=\frac{\partial f(x)}{\partial x} \quad (5)$$

If f is the Sigmoid function, and

$$f^{'}(x)=\frac{e^{-x}}{(1+e^{-x})^2}=f(x)(1-f(x)) \quad (6)$$

The update rule for the weights from the hidden layer to the output layer is

$$w_{km(new)}^{[2]}=w_{km(old)}^{[2]}+\eta\,\Delta w_{km}^{[2]} \quad (7)$$

where $k=1,...,n_H$, $m=1,...,n_O$ and η is the learning rate. The update rule for the weights from the input layer to the hidden layer is

$$\Delta w_{km}^{[1]}=x_l f^{'}(y_k)\sum_{m=1}^{n_O} w_{km}^{[2]} f^{'}(z_m)(z_m-d_m) \quad (8)$$

$$w_{lk(new)}^{[1]}=w_{lk(old)}^{[1]}+\eta\,\Delta w_{lk}^{[1]} \quad (9)$$

where $l=1,...,n_l$, $k=1,...,n_H$.

III. Electromagnetic Field Computation

In this study, the magnetic field is calculated using the FEM. This method is based on the magnetic vector potential $\boldsymbol{A}$ representation of the magnetic field [13]. The calculations are performed in two steps. First, the magnetic field intensity is calculated by solving the system of equations:

$$\boldsymbol{rot}(\boldsymbol{E})=-\frac{\partial \boldsymbol{B}}{\partial t} \quad (10)$$

$$\boldsymbol{rot}(\boldsymbol{H})=\boldsymbol{J} \quad (11)$$

$$div(\boldsymbol{B})=0 \quad (12)$$

where $\boldsymbol{H}$ and $\boldsymbol{E}$ are the magnetic and electric field respectively, $\boldsymbol{B}$ the magnetic induction and $\boldsymbol{J}$ the electric current density. This system of equations is coupled with relations associated to material property, material being assumed to be isotropic:

$$\boldsymbol{B}=\mu(|\boldsymbol{H}|)\boldsymbol{H} \quad (13)$$

$$\boldsymbol{J}=\sigma\boldsymbol{E} \quad (14)$$

where μ is the magnetic permeability, σ is the electrical conductivity.

The magnetic vector potential $\boldsymbol{A}$ is expressed by

$$\boldsymbol{B}=\boldsymbol{rot}(\boldsymbol{A}) \quad (15)$$

The electromagnetic field analysis for a cartesian system is carried out by the FEM [14]. The equation of the electromagnetic field is expressed by $\boldsymbol{A}$ as

$$\boldsymbol{rot}\left(\frac{1}{\mu}\boldsymbol{rot}\,\boldsymbol{A}\right)+\sigma\frac{\partial \boldsymbol{A}}{\partial t}=\boldsymbol{J}_s \quad (16)$$

where $\boldsymbol{J}_s$ is the vector of supply current

Equation (16) is discretized using the Galerkin FEM, which leads to the following algebraic matrix equation

$$([\boldsymbol{K}]+j\omega[\boldsymbol{C}])[\boldsymbol{A}]=[\boldsymbol{F}] \quad (17)$$

with:

$$\boldsymbol{A}=\sum\alpha_j(x,y)\boldsymbol{A}_j \quad (18)$$

α_j is the interpolation function.

$$\boldsymbol{K}_{ij}=\iint_{\Omega}\frac{1}{\mu}\,\boldsymbol{grad}\,\alpha_i\;\boldsymbol{grad}\,\alpha_j\;dxdy \quad (19)$$

$$\boldsymbol{C}_{ij}=\iint_{\Omega}\sigma\,\alpha_i\alpha_j\;dxdy \quad (20)$$

$$\boldsymbol{F}_i=\iint_{\Omega}\boldsymbol{J}_s\,\alpha_i\;dxdy \quad (21)$$

α_i is the projection function.

In the second step, the field solution is used to calculate the magnetic induction $\boldsymbol{B}$. More details about the finite element theory can be found in [14].

IV. Methodology For Parameters Identification

First of all, an electromagnetic device was idealized to be used as an electromagnetic field exciter (Fig. 3). In this paper, we have considered direct current in the coils. To increase the sensitivity of the electromagnetic device a magnetic core with a high permeability is used. Deviations of the magnetic induction (difference in magnetic induction without and with material under test) at equally stepped points in the external surface of the material under test are taken (Fig. 4).

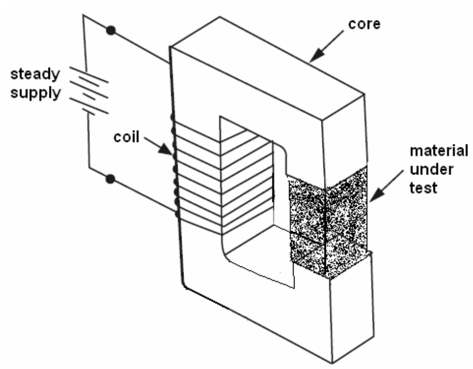

Fig. 3 Structure of electromagnetic field exciter

The methodology used in this work is summarized in the following steps. Steps 1-3 correspond to the finite element analysis:

1) Generation of the initial finite element mesh.
2) Modifications in the physical property, changing relative magnetic permeability and electrical conductivity of the material under test.
3) Finite element solution getting the magnetic inductions values at the sensor position.
4) Generation of the neural network training vectors
5) Definition of the neural network architecture and training vectors.
6) Neural network training, Validation tests and identification of new parameters.

The problem was solved on a PC with P4 2.4G CPU under Matlab® 6.5 workspace using the Partial Differential Equation Toolbox and Neural Network Toolbox for the finite element meshes generation and neural networks architecture definition respectively [15], [16]. For the finite element problem resolution and the inverse problem solution, we use programs developed by us.

The simulations were done for a hypothetic metallic wall with 2 mm height and 12 mm width. The material under test is 1006 Steel (a magnetic material). The relative magnetic permeability of the core is supposed to be 4000.

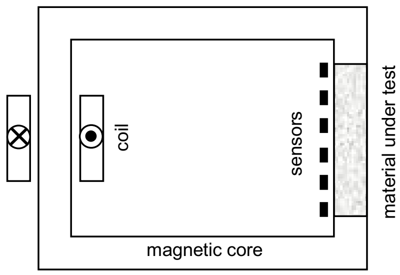

Fig. 4 Arrangement for the measurements

During the phase of finite elements simulations, errors can appear, due to it's massively nature. So, the results of the simulations must be carefully analyzed. This can be done, for instance, plotting in the same graphic the magnetic induction deviations for a set of parameters. Fig. 5 shows the magnetic induction deviation in the region of the device at the sensor position for three materials having the same electrical conductivity (6 10^3 [S/m]), and relative magnetic permeability ranging from 185 to 650. A similar graphic, with electrical conductivity equal to 2 10^7 [S/m] and magnetic relative permeability ranging from 185 to 650 is shown in Fig. 6. Fig. 7 shows the graphics for a fixed magnetic relative permeability (320), and three different electrical conductivity ranging from 2 10^5 [S/m] to 1.5 10^8 [S/m]. Fig. 8 shows a similar graphic, for the magnetic relative permeability equal to 560.

The coherence of the curves in these graphics allows us to infer if there are or not errors in the dataset.

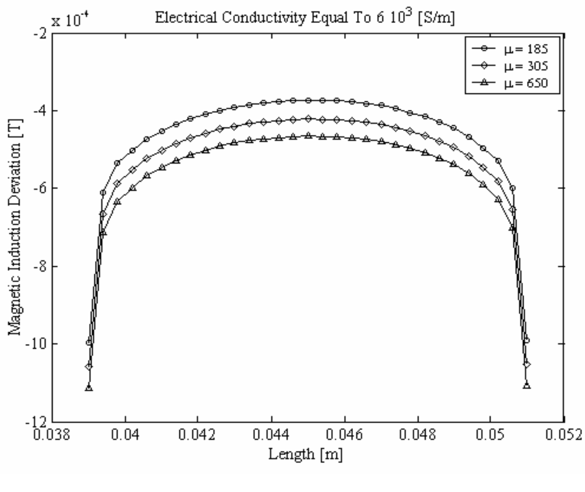

Fig. 5 Magnetic induction deviation for three values of magnetic relative permeability and electrical conductivity equal to 6 10^3 (S/m)

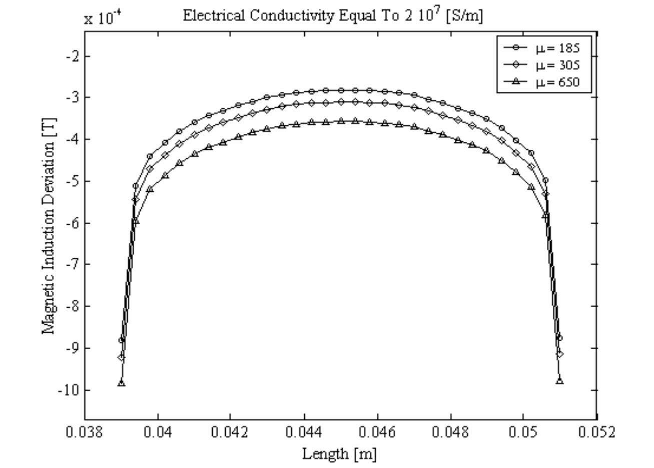

Fig. 6 Magnetic induction deviation for three values of magnetic relative permeability and electrical conductivity equal to 2 10^7 (S/m)

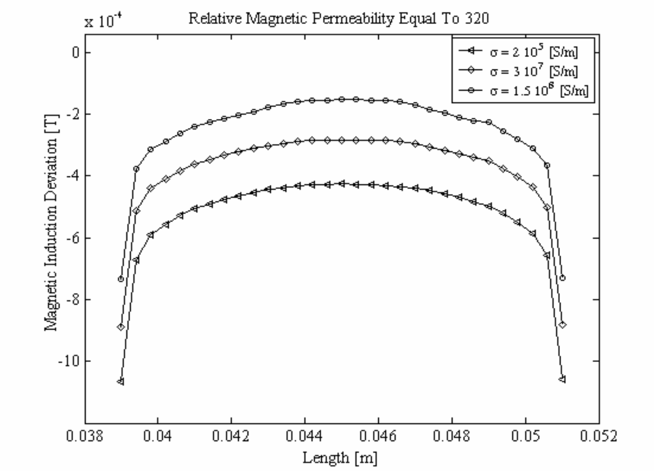

Fig. 7 Magnetic induction deviation for three values of electrical conductivity and magnetic relative permeability equal to 320

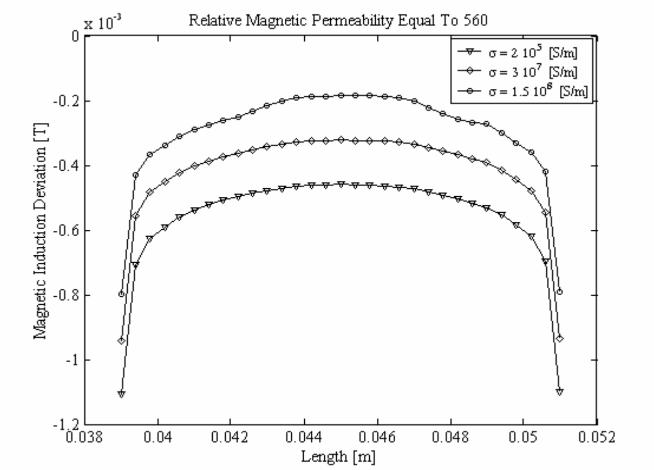

Fig. 8. Magnetic induction deviation for three values of electrical conductivity and magnetic relative permeability equal to 560

V. Formulation Of Network Models

In the step 4, we generate the training vectors for neural networks. In this work, we generated 300 vectors for neural networks training. Each of the vectors consists of 16 input values, which represent the deviation of magnetic induction, and two output values, which represent the relative magnetic permeability and electrical conductivity of the material under test. Of the 300 vectors, a random sample of 225 cases (75 %) was used as training, 75 (25 %) for validation. Training data were used to train the application and the validation data were used to monitor the neural network performance during training.

To show stability of the proposed approach, the measured values, which intrinsically contains errors in the real word, is obtained by adding a random perturbation to the exact inputs values of the network, such that

$$\tilde{I}n = In_{exact} + \sigma\lambda \quad (22)$$

where σ is the standard deviation of the errors and λ is a random variable taken from a Gaussian distribution, with zero mean and unitary variance.

Twin numerical experiments were performed. In the first one, noiseless data where employed $(\sigma = 0)$. The second numerical experiment was carried out using 5 % of noise $(\sigma = 0.05)$.

The MLP neural network architecture considered for this application was a single hidden layer with sigmoid activation function. The learning rate initially is 0.1 but as the root mean squared error gets smaller it decreases to 0.01. This is the experience from the training which also matches the idea of learning rate annealing in [7].

A back-propagation algorithm based on Levenberg-Marquardt optimization technique [17] was used to model MLP for the above data.

The Levenberg-Marquardt technique was designed to approach second order training speed without having to compute the Hessian matrix [17]. This matrix approximated with use of the Jacobian matrix which can be computed through a standard back propagation algorithm that is much less complex than computing the Hessian matrix. The performance function will always be reduced on each iteration of the algorithm.

For the MLP neural network, several network configurations were tried, and better results have been obtained by a network constituted by one hidden layers with 28 neurons. The MLP architecture had 16 input variables, one hidden layer and two output nodes. Total number of weights present in the model was 534. The best MLP was obtained at lowest mean square error of 0.00057. Percentage correct prediction of the MLP model was 96.4 % and 94.7 % for noiseless and noise data respectively.

Fig. 9 shows the performance of the MLP neural network during a training session. Table 1 show some results for the validation of the network, for this session.

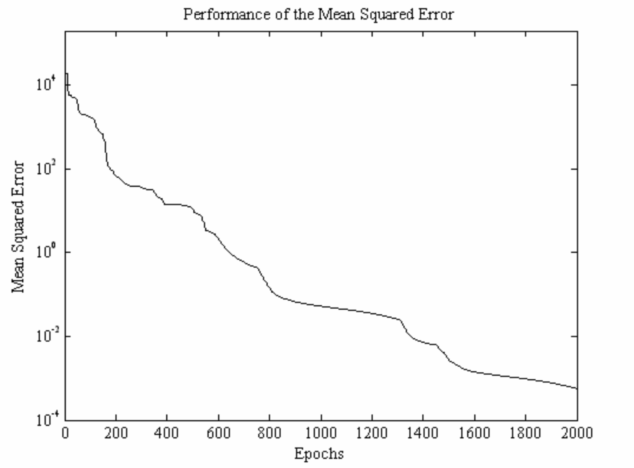

Fig. 9 Performance of the MLP network during a training session

TABLE I
EXPECTED AND OBTAINED VALUES DURING A TRAINING SESSION

Relative magnetic permeability			Electric conductivity		
Expected	Obtained		Expected	Obtained	
	0 % Noise	5 % Noise		0 % Noise	5 % Noise
160.05	160.21	159.23	$1.700\ 10^1$	$1.704\ 10^1$	$1.712\ 10^1$
247.89	247.87	246.80	$3.500\ 10^2$	$3.506\ 10^2$	$3.513\ 10^2$
321.19	321.21	320.63	$4.200\ 10^6$	$4.211\ 10^6$	$4.218\ 10^6$
448.17	448.25	449.08	$3.000\ 10^5$	$3.010\ 10^5$	$3.016\ 10^5$
526.12	526.28	527.01	$7.500\ 10^3$	$7.509\ 10^3$	$7.512\ 10^3$
640.16	640.33	639.76	$2.210\ 10^1$	$2.215\ 10^1$	$2.222\ 10^1$

As we can see, the results obtained in the validation are very close to the expected ones. The worse identification was obtained by MLP network with noises data.

VI. New parameter identification

After the neural networks training and respective validations, new parameters were simulated by the FEM, for posteriori identification by the networks. Table 2 shows the values of electromagnetic parameters, and the obtained values, by the neural networks.

TABLE II
SIMULATION RESULTS FOR NEW PARAMETERS

Parameter	Relative magnetic permeability			Electrical conductivity		
	Expected	Obtained		Expected	Obtained	
		0 % Noise	5 % Noise		0 % Noise	5 % Noise
1	87.00	87.11	86.18	$6.410\ 10^1$	$6.415\ 10^1$	$6.506\ 10^1$
2	214.00	214.20	213.14	$2.750\ 10^2$	$2.767\ 10^2$	$2.773\ 10^2$
3	368.00	368.27	367.20	$2.230\ 10^6$	$2.254\ 10^6$	$2.258\ 10^6$
4	476.00	476.31	475.19	$4.160\ 10^5$	$4.179\ 10^5$	$4.182\ 10^5$

As we can see, the results obtained in the identification of new parameters, obtained by the neural networks agree very well with the expected ones.

VII. Conclusion

In this paper we presented an investigation on the use of the FEM and MLP neural network for the identification of parameters in metallic walls. This study indicates the good and stable predictive capabilities of MLP neural network in the presence of noise.

The association of FEM and ANN techniques seems to be a useful alternative for identification of parameters trough inverse analysis. Future works are intended to be done in this field, such as the use of more realistic FEM, computer parallel programming, in order to get quickly solutions.

References

[1] P. Ramuhalli, Neural network based iterative algorithms for solving electromagnetic NDE inverse problems, Ph.D. dissertation, Dept. Elect. Comp. Eng, Iowa Univ, USA, 2002.
[2] E. Coccorese, R. Martone and F. C. Morabito, A neural network approach for the solution of electric and magnetic inverse problems, *IEEE Trans. Magnetics*, vol. 30, no. 5, pp. 2829-2839, 1994.
[3] S. R. H. Hoole, Artificial neural networks in the solution of inverse electromagnetic field problems, *IEEE Trans. Magnetics*, vol. 29, no 2, pp. 1931-1934, 1993.
[4] P. Ramuhalli, L. Udpa and S.S. Udpa, Finite element neural networks for solving differential equations, *IEEE Trans. Neural Networks*, vol. 16, no. 6, pp. 1381-1392, 2005.
[5] P. M. Wong and M. Nikravesh, Field applications of intelligent computing techniques, *J. Petrol Geolog*, vol. 24, no. 4, pp. 381-387, 2001.
[6] A. Fanni and A. Montisci, A neural inverse problem approach for optimal design, *IEEE Trans. Magnetics*, vol. 39, no 3, pp. 1305-1308, 2003.
[7] S. Haykin, Neural networks: A comprehensive foundation, Englewood Cliffs, NJ: Prentice-Hall, New York, 1999.
[8] I. V. Turchenko, Simulation modelling of multi-parameter sensor signal identification using neural networks, *Proc. 2nd IEEE Int Conf. Intelligent Systems, Bulgaria, 2004, pp. 48-53.*
[9] N.P. De Alcantara, J. Alexandre, M. De Carvalho, Computational investigation on the use of FEM and ANN in the non-destructive analysis of metallic tubes, *Proc. 10th Biennial Conf. Electromagnetic Field Computation*, *Italy, 2002.*
[10] A. K. Jain, J. Mao and K. M. Mohiuddin, Artificial neural networks: a tutorial, *Computer*, pp. 31-44, 1996.
[11] K. Mehrotra, C. K. Mohan and S. Ranka, *Elements of artificial neural networks*, MA: MIT Press, Cambridge, 1997.
[12] D. Cherubini, A. Fanni, A. Montisci, P. Testoni, A fast algorithm for inversion of MLP networks in design problems, *COMPEL. Int. J. Comp and Math in Electric and Electro Eng*, vol. 24, no. 3, pp. 906-920, 2005.
[13] M. V. K. Chari and S. J. Salon, *Numerical methods in electromagnetism*, CA: Academic, San Diego, 2000.
[14] P.P. Silvester and R.L. Ferrari, *Finite elements for electrical engineers*, Univ Press, Cambridge, 1996.
[15] Z. Raida, Modeling EM tructures in the neural network toolbox of MATLAB, *IEEE Antenna's and propagation Magazine*, vol. 44, no. 6, pp. 46-67, 2002.
[16] Partial Differential Equation Toolbox user's guide, for use with MATLAB, The Math Works Inc.
[17] M. T. Hagan and M. Menhaj, Training feed-forward networks with the Levenberg-Marquardt algorithm, *IEEE Trans Neural Networks*, vol. 5, no. 6, pp. 989-993, 1994.

T. Hacib was born in Jijel, Algeria in 1977. he recieved the Licence degree in Electrical Engineering in 2001, and the Magister in Electrical Engineering in 2004 from the Electrical Engineering Institute of Jijel University, Algeria. He is currently working toward the Ph.D degree at Jijel University. His principal research interests include inverse problems, computations of electromagnetic field and neural networks applications in identification of parameters and defects of electromagnetic systems. Mr Hacib has participated in many international conferences and he his published in more than 10 works between conferences proceedings and journals. He is member of LAMEL laboratory.

M.R. Mekideche was born in Jijel, Algeria in 1957. After his first diploma as electrical engineer at the USTO University at Oran, he joined, in 1981, the National Polytechnicas School at Algiers as an assistant and post-graduate student preparing a Magister. He got the "Maître assistant" degree in 1986. After his PhD degree from the University of Nantes (France) he became "Maître de Conferences" and then Professor. He is now at the University of Jijel, as a teacher and Dean of the engineering faculty.
Prof. Mekideche is now heading the LAMEL laboratory of the University of Jijel and is the author or co-author of about 50 works published or presented at international conferences.

N. Ferkha was born in Jijel, Algeria in 1977. She recieved the engineer degree in Electrical Engineering in 2000, and the Magister in Electrical Engineering in 2004 from the Electrical Engineering Institute of Jijel University, Algeria, where she has now an Ph.D student. Here current research interests include vibratory phenomena, computations of electromagnetic field and neural networks applications in shape optimization of electromagnetic systems. She is member of LAMEL laboratory.