J Supercond Nov Magn (2011) 24: 683–687
DOI 10.1007/s10948-010-0937-z

ORIGINAL PAPER

Phase Identification of La-Doped Hard Magnetic Barium Ferrite Using Artificial Neural Network

Huseyin Sozeri · Ilker Kucuk · Husnu Ozkan

Received: 13 September 2010 / Accepted: 14 September 2010 / Published online: 6 October 2010

Abstract A model based on an artificial neural network (ANN) was designed for the simulation and estimation of 2 theta and intensity values obtained by X-Ray Diffraction (XRD) of pure and La-doped barium ferrite powders which have been synthesized in ammonium nitrate melt. Its performance is evaluated by the influences of different La content, sintering temperature, Fe/Ba ratio, and washed in HCl (or not washed in HCl) samples. The XRD patterns of samples estimated by the ANN agree well with the experimental values, indicating that the model is reliable and adequate.

Keywords La doped · Barium ferrites · Hard magnetic materials · Modeling · Neural network

1 Introduction

The barium ferrite has attracted great interest in the magnetic material field and has been widely used as a permanent magnet because of its fairly large magnetocrystalline anisotropy and high Curie temperature, together with relatively large saturation magnetization, excellent chemical stability, and corrosion resistivity. Various chemical routes were used to synthesize nanoparticles of Ba-hexaferrite: microemulsion [1], chemical co-precipitation [2], glass crystallization [3], combustion, sol–gel synthesis [4], hydrothermal synthesis [5], and in a nitric acid [6]. The ammonium nitrate melt technique is based on dissolution of initial oxides in ammonium nitrate. The method is simple, effective, and fast [7].

The Fe:Ba ratio is important to avoid formation of impurity phases like hematite (α-Fe_2O_3). Besides, to eliminate the non-magnetic phase, the annealed powders can be washed with hydrochloric acid (HCl).

Learning and generalization ability, real-time operation, and ease of implementation have made ANNs popular in the last years [8]. ANNs have been applied in many areas because of these features [9–11]. The available ANN software today provides many neural-network architectures and learning algorithms, and also helps users to apply ANN to their specific problems easily.

In this work, the 2 theta and intensity values measured by XRD in barium ferrite samples are modeled by using the Fe/Ba ratio, La content, sintering temperature, HCl washing, and a set of our unpublished experimental data [12]. The developed model is more accurately and easily predicted.

2 Experimental

Appropriate amounts of $BaCO_3$, Fe_2O_3 and La_2O_3 powders of high purity (99%) were weighed and mixed at different La content (0.0, 0.1, 0.2, 0.3, 0.4, 0.6, 0.8, and 1.0) and different Fe/Ba atomic ratios (2, 4, 6, 8, 10, 10.5, 11, 11.5, and 12). The mixed powder was put into the melted ammonium nitrate. The solution was mixed on a hot plate with magnetic stirrer until the liquid disappears. Before grinding in an agata mortar for 15 min, a small amount of isopropyl alcohol was

H. Sozeri
TUBITAK-UME, National Metrology Institute, PO Box 54, 41470 Gebze-Kocaeli, Turkey

I. Kucuk (✉)
Physics Department, Faculty of Arts and Sciences, Uludag University, Gorukle Campus 16059, Bursa, Turkey
e-mail: ikucuk@uludag.edu.tr

H. Ozkan
Department of Physics, Middle East Technical University, 06531 Ankara, Turkey

added to make wet grinding. The precursor was calcinated at 450 °C for 5 h to remove possible organic compounds. Then the precursor was again ground for 15 min with isopropyl alcohol and dried and pelletized under the pressure of 200 MPa. The pressed disc-shaped pellets (diameter 16 mm, thickness 3 mm) were sintered at 800–1200 °C for 2 h in the air and then cooled in the furnace. Finally, the powders were washed with HCl.

Structural analysis was done by Rigaku–Minflex diffractometer (Cu-K_α) utilizing the ICDD qualitative analysis software. X-ray powder diffraction diagrams were obtained in the 2–70° 2θ range with a step range of 0.01° and a measuring time of 0.3 s per step. The magnetic properties of samples were examined at room temperature using vibrating sample magnetometer (LDJ Electronics Inc., Model 9600) with the maximum field up to 1.6 T.

3 Artificial Neural Network

A neural network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the human neuron [13]. The processing ability of the network is stored in the inter-unit connection strengths or weights, obtained by a process of adaptation to, or learning from, a set of training patterns. A neural network model usually assumes that the computation is distributed over several processing units which are interconnected and which operate in parallel. The most popular network is multi-layer perception, which is a feed forward network, i.e., all signals flow in a single direction from the input to the output of the network. The error between the actual and expected outputs is used to modify the strengths, or weights, of the connections between the neurons. This method of training is known as supervised training. In multi-layer perceptions, the back-propagation algorithm for supervised training is often adopted to propagate the error from the output neurons and compute the weight modifications for the neurons in the hidden layers. In an untrained network, an input vector $\vec{X}$ yields an output vector $\vec{Z}$, which usually differs substantially from the target output. The neural network is fed with a series of vectors $\vec{X}$ for which corresponding target $\vec{Z}_T$ has been determined by means of model core experiments. Differences between $\vec{Z}$ and $\vec{Z}_T$ are evaluated in order to modify the weights in a way that reduces the difference iteratively.

The error for any hidden or output node k is given by

$$\Delta w_{ki} = \alpha \delta_k x_{ki}^p \tag{1}$$

where α and x_{ki}^p are the learning ratio of the network, the kth input data, respectively. δ_k is the error of the kth neuron in the hidden layer. Where hidden nodes,

$$\delta_k = \sigma a_k \sum_{j=I_k} \delta_j w_{jk} \tag{2}$$

and for output nodes,

$$\delta_k = \sigma a_k \left(t_k^p - y_k^p\right) \tag{3}$$

σ is transfer function, a_k is the weight of the kth neuron, t_k^p is the target data of the kth neuron.

4 Genetic Algorithm

The genetic algorithm works with an initial population which may correspond to numerical values of a particular variable [14]. The size of initial population is usually associated with the problem under consideration. The members of this population are usually strings of zeros or ones and they are encoded values of a variable or variables to be examined. This initial population is generated randomly and the terminology of genetics can be used for characterizing it. Each string in the population corresponds to a chromosome and each binary element of the string to a gene. A new population is developed from this initial population using the analogue of specific fundamental genetic processes which are reproduction based on fitness, crossover, and mutation.

The performance of the optimization model was calculated by the fitness function based on least mean square error (LMSE). LMSE for the network outputs (for 2θ) was calculated by

$$e_1 = \sum_{k=1}^{n} \left[(2\theta)_{d,k} - (2\theta)_{c,k}\right]^2 \tag{4}$$

where n is the population size, $(2\theta)_{d,k}$ and $(2\theta)_{c,k}$ are the desired and the calculated 2θ. The total fitness value was calculated by taking the inverse of the averaged LMSE [14].

5 Levenberg–Marquardt Algorithm

Neural network training is usually formulated as a nonlinear least-squares problem. Essentially, the Levenberg–Marquardt algorithm is a least-squares estimation algorithm based on the maximum neighborhood idea. Let $E(w)$ be an objective error function made up of m individual error terms $e_i^2(w)$ as follows:

$$E(w) = \sum_{i=1}^{m} e_i^2(w) = \|g(w)\|^2 \tag{5}$$

Table 1 Predicted and measured the 2θ and intensity values for barium ferrites used as training data

Barium Ferrite Composition	Fe/Ba	T (°C)	$BaFe_{12}O_{19}$				α-Fe_2O_3			
			2θ (°)		Intensity (a.u.)		2θ (°)		Intensity (a.u.)	
			Measured	ANN	Measured	ANN	Measured	ANN	Measured	ANN
$Ba_{0.6}La_{0.4}Fe_{12}O_{19}$	2 (washed in HCl)	1000	30.32	30.38	152	167				
			32.14	32.22	274	271				
			34.10	34.17	302	317				
			37.08	37.15	146	161				
			40.31	40.39	102	104				
			55.03	55.03	134	127				
			56.53	56.59	142	143				
			63.06	63.19	134	149				
	10	1000	30.36	30.34	150	152	33.16	33.03	100	94
			32.19	32.25	366	356	35.68	35.58	94	89
			34.14	34.12	300	330				
			37.13	37.20	132	120				
			55.06	55.17	108	119				
			56.62	56.73	128	141				
			63.10	63.23	130	143				
$Ba_{0.7}La_{0.3}Fe_{12}O_{19}$	12	800	30.31	30.37	85	94	24.30	24.79	80	85
			32.22	32.28	102	112	33.28	33.95	355	343
			34.24	34.31	102	112	35.73	36.44	300	291
			55.04	55.10	80	88	49.57	48.58	110	120
			63.04	63.07	82	90	54.16	55.24	250	209
							62.52	63.77	103	115
							64.15	62.74	110	117
$BaFe_{12}O_{19}$	12	1200	30.39	30.45	145	160				
			32.21	32.29	250	250				
			34.18	34.25	305	320				
			37.12	37.19	137	151				
			55.04	55.15	95	97				
			56.57	56.57	156	148				
			63.04	63.10	131	132				
$Ba_{0.8}La_{0.2}Fe_{12}O_{19}$	12	1000	30.31	30.24	94	90	33.18	33.01	56	55
			32.18	32.26	152	148	35.62	35.58	48	45
			34.14	34.20	148	140	54.05	53.98	44	48
			37.05	36.93	92	95				
			40.32	40.41	76	70				
			55.12	55.20	74	80				
			56.52	56.59	96	90				
			63.11	63.05	84	95				

where $e_i^2(w) = (y_{di} - y_i)^2$, y_{di} is the desired value of output neuron i, y_i is the actual output of that neuron and $g(w)$ is a function containing the individual error terms. The aim of the Levenberg–Marquardt algorithm is to compute the weight vector w such that $E(w)$ is minimum. Using the Levenberg–Marquardt algorithm, a new weight vector w_{k+1} can be obtained from the previous weight vector w_k as follows:

$$w_{k+1} = w_k + \delta w_k \tag{6}$$

where δw_k is defined as

$$\delta w_k = -\left(J_k^T g(w_k)\right)\left(J_k^T J_k + \lambda I\right)^{-1} \tag{7}$$

In (7), J_k is the Jacobian of $g(w_k)$ evaluated by taking derivative of $g(w_k)$ with respect to w_k, λ is the Marquardt parameter, I is the identity matrix [15].

6 ANN Applications to XRD of Hard Magnetic Barium Ferrites

A total of 68,000 input vectors were available in the training data set of experimental data for 35 barium ferrite samples with different sintering temperatures, Fe/Ba ratios, and La contents. Because of the large number of input vectors considered, a sub-network has been developed for each sintering temperature and La-content. This provides rapid network development and enhances accuracy. The sub-networks are integrated to a single database. A set of data was used for testing the network. Matlab™ has been used for training the networks, giving the advantage of rapid network development through flexible choices of algorithms, output functions, and other training parameters, thereby enhancing accuracy. A multi-layered neural network, which has 4 input neurons, 2 output neurons, 8 neurons of hidden layers, and full connectivity between neurons, was developed. The number of hidden layers and the number of neurons in hidden layers were determined through trial and error to be optimal. The input parameters were the Fe/Ba ratio, La content (x), sintering temperature (T) and HCl washing are given. The output parameters were 2θ and intensity of XRD. The network has been trained by using the genetic and Levenberg–Marquardt algorithm with the hyperbolic tangent transfer function in hidden layer, sigmoid transfer function in output layer and 1,000 epochs. After the network was trained, the mean squared error was approximately found to be 0.0005. When the network has been tested with the training data set, the linear correlation coefficient was found to be on average 99% for the 2θ and intensity.

Table 1 compares predictions and experimental data of the 2θ and intensity for pure and impurity phases a set of samples. These alloys are within the training data set. The average absolute difference for the predictions and experimental data is about 2.5%. This shows the ANN has an acceptable prediction capability for our samples within the training data.

Figure 1 shows the 2θ values versus the network outputs for training data set. The diagonal line in this graph shows perfect match between measurement and network output. Figure 2 also compares between prediction of ANN and ex-

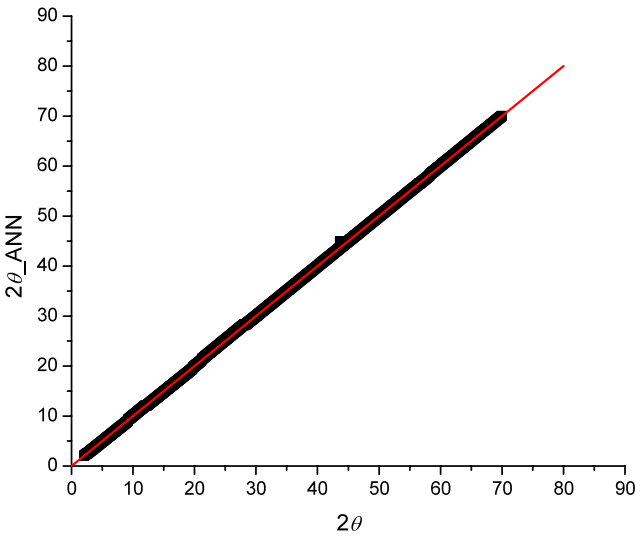

Fig. 1 Plot of the measurement 2θ versus the network outputs for $Ba_{0.6}La_{0.4}Fe_{12}O_{19}$ with Fe/Ba = 12

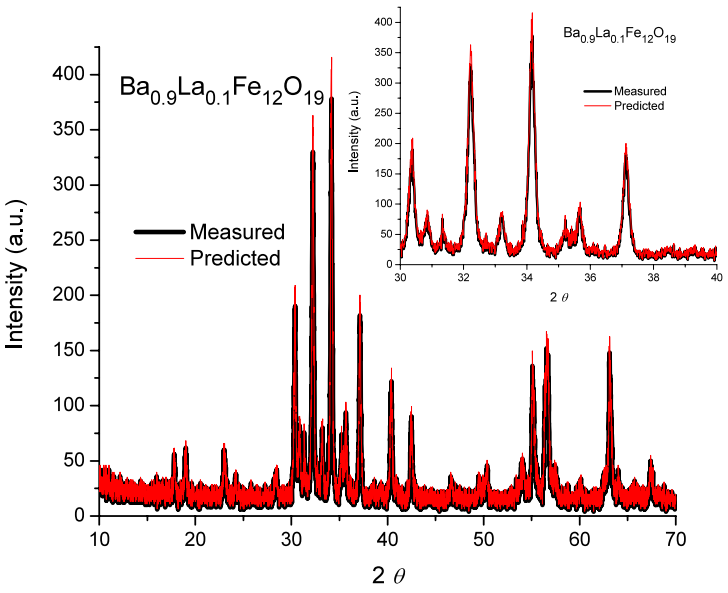

Fig. 2 Comparison of predicted and measured XRD patterns for untrained data

perimental 2θ and intensity for $Ba_{0.9}La_{0.1}Fe_{12}O_{19}$ sample which is not include to training data. The correlation is approximately found to be 99%.

The proposed method has some inherent limitations which makes it not a general solution. The trained neural network is based on a specific set of chemical composition of hard magnetic barium ferrite. That is to say, the trained network can be only valid for the same chemical compositions. For very different chemical compositions, a series of experiments would have to be performed again to obtain input vectors for a new ANN or the proposed ANN training. If the ANN input vectors could include the XRD information of new barium ferrite powders, many more experimental data would have to be done to meet the accuracy requirements for a more general solution. However, since the neural model presented in this work has accuracy and requires no tremendous computational efforts and less background information about the barium ferrites, it can be used to predict more accurately and easily the XRD patterns of these samples.

7 Conclusion

The ANN has been successfully applied for the prediction of the XRD patterns of pure barium ferrite and La-doped barium ferrite. The results are found to be in very good agreement with our experimental data. The developed ANN model can be used for the prediction of phase identification of hard magnetic barium ferrites which are synthesized by the different preparation conditions.

References

1. Lui, X., Wang, J., Gan, L.-M., Ng, S.-C., Ding, J.: J. Magn. Magn. Mater. **184**, 344 (1998)
2. Jacobo, S.E., Domingo-Pascal, C., Rodriguez-Clemente, R., Blesa, M.A.: J. Mater. Sci. **32**, 1025 (1997)
3. Muller, R., Heirgeist, R., Steinmetz, H., Ayoub, N., Fujisaki, M., Schuppel, W.: J. Magn. Magn. Mater. **201**, 34 (1999)
4. Huang, J., Zhuang, H., Li, W.L.: Mater. Res. Bull. **38**, 149 (2003)
5. Liu, X., Wang, J., Gan, L.M., Ng, S.C.: J. Magn. Magn. Mater. **195**, 452 (1999)
6. Sözeri, H.: J. Magn. Magn. Mater. **321**, 2717 (2009)
7. Sözeri, H., Ghazanfari, N.: Mater. Chem. Phys. **113**, 977–981 (2009)
8. Haykin, S.: Neural Networks: A Comprehensive Foundation. Macmillan, New York (1994)
9. Kucuk, I.: J. Magn. Mag. Mater. **305**, 423 (2006)
10. Kucuk, I.: J. Magn. Mag. Mater. **307**, 53 (2006)
11. Kucuk, N.: Ann. Nucl. Energy **35**, 1787 (2008)
12. Kucuk, I., Sözeri, H., Ozkan, H.: (submitted)
13. Pham, D.T., Liu, X.: Principles of Artificial Neural Network. World Scientific, Singapore (1997)
14. Goldberg, D.E.: Gene. Algorithms search. Optim. Mach. Learn. Reading (1989)
15. Wang, F., Devabhaktuni, V.K., Xi, C., Zhang, Q.: Int. J. RF Microw. Comput.-Aided Eng. **9**, 216 (1999)