American Journal of Electromagnetics and Applications
2014; 2(3): 27-33
Published online September 10, 2014 (http://www.sciencepublishinggroup.com/j/ajea)
doi: 10.11648/j.ajea.20140203.12

New two variable search algorithm implemented for Preisach model

Tabara, Octavian Adrian

Dept. of electrical Engineering, University "Politehnica" of Bucharest, Bucharest, Romania

Email address:
tavi.tabara@gmail.com

To cite this article:
Tabara, Octavian Adrian. New Two Variable Search Algorithm Implemented for Preisach Model. *American Journal of Electromagnetics and Applications.* Vol. 2, No. 3, 2014, pp. 27-33. doi: 10.11648/j.ajea.20140203.12

Abstract: Preisach model is the most used hysteresis model in magnetic materials research. This article presents problems that appear in implementation of a two variable search algorithm for Preisach model identification. The algorithm is use to find the optimal parameters values for the probability function distribution. The tests for a subway card in the case of longitudinal and transversal magnetization are available.

Keywords: Hysteresis, Preisach, Identification

1. Introduction

Hysteresis is a complicated phenomenon. As a general definition, it represents a relation between input and output, taking into account the input history [1]. In order to characterize magnetic materials hysteresis models were develop. Many authors studied the subject of hysteresis in magnetic materials and a description from a mathematical point of view of this phenomenon is a difficult task [2-6]. In order to characterize hysteresis, numerical models were developing. One of the first developed hysteresis models, based on an idea of Weiss and de Freudenreich, is the classical Preisach model [7]. Many rectangular elementary particles called hysterons model the material. Another basic hysteresis model is the Stoner-Wohlfarth model [8]. This model considers the magnetization and the applied magnetic field in different directions. Because of this, it is consider the first developed vectorial model. In this model, particles are ellipsoidal. Development of hysteresis models is an actual subject in scientific literature. Many papers contain a state of the art of developed hysteresis models [9, 10].

These models are the basic models in hysteresis. Starting from basic ideas of these fundamental models, numerous improvements are available. Because these models consider that magnetization is not influence by interactions between particles, numerous improvements are proposed.

For example in case of Stoner-Wohlfarth models, Atherton and Beattie supposed the existence of an average interaction field. This additional hypothesis complicates identification procedure. Time needed for identification increases compared to the classical Stoner-Wohlfarth model. It works well only in certain cases as the authors proved in their paper [11].

For Preisach hysteresis model, one of the first extensions that suggest the existence of an average interaction field is the moving model [12]. In this case, the interaction field is proportional with magnetization. Implementation of this model in a scientific software for calculation proved that it requires a lot of time for identification [13].

Properties of a magnetic material define with the help of the probability density function. This function expresses the probability for an elemental particle to have specific tipping threshold. Numerical identification of this distribution raised problems and distributions have been proposing for characterization of particles along each axis in different magnetic materials [14].

This article presents a two variable search algorithm, which is use to find 2 optimal unknown values for any analytical probability distribution, starting from a reduced set of experimental data. The identification of probability function can be used in electrical engineering design, as it is proved by the tests done for a subway card in case of longitudinal and transversal magnetization. The codes can also a built in library for different software that treat hysteresis problem.

2. Preisach Model

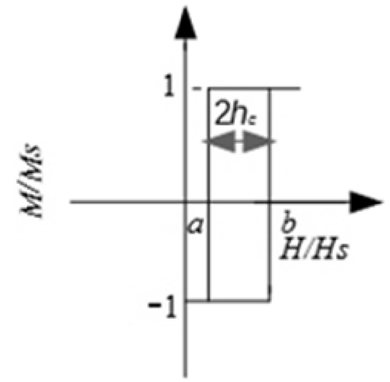

Figure 1. *Hysteresis cycle of a hysteron*

A popular hysteresis model in electrical engineering, it is represent by classical Preisach model [15-17]. It assumes that many elementary particles can describe material with rectangular asymmetric cycle called hysterons. The hysteresis cycle of a hysteron is available in Figure 1.

Hysteresis. In Figure 1, a and b represent the tipping thresholds values, h_c represents the coercive reported field intensity M represents magnetization (output), M_s represents the saturation magnetization, H represents applied field (input) and H_s is the saturated value of applied field. Essential property of hysteresis phenomenon is represented by history saving or memory. In Preisach model history is consider through state line [18]. The initial application of field determines positive and negative elementary particles. In the Preisach model, the evolution of the state line follows this rule: if the current applied field is bigger than the previous value of field the state line is going towards right; else the state line is going down [19]. The application of a magnetic field determines zones that are pass by this field to change sign. For a faster calculation, in the code only these areas where considered.

The magnetization is determined using equation (1):

$$M = M_S \left(\iint\limits_{S+} p(a,b)dadb - \iint\limits_{S-} p(a,b)dadb \right) \tag{1}$$

In equation (1), M represents calculated magnetization by model, M_s represents saturation magnetization, S_+, S_- are corresponding areas of positive, negative hysterons and $p(a, b)$ is probability density function used in Preisach model [20].Main advantages of Preisach model are simplicity and speed in describing hysteresis process. One important disadvantage is that no physical background of this model exists because it is a purely phenomenological model.

This article proposes an estimation of hysteresis using a numerical calculation taking into account speed. It can be apply to engineering problems where the balance between used time in simulations and precision is very important. In addition, numerical problems from implementation are detail for a better understanding of implementation ideas.

3. Implementation of Preisach Model

Figure is as follows: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence. Matlab is a powerful programming environment used in electrical engineering applications. It is use in numerical calculus and allows easy operations with graphical results. It can also be utilize in implementation and development of new models or applications [21].

Fundamental problem in magnetic hysteresis modeling it is represented by an accurate description, from a mathematical point of view, of probability density function. Numerous analytic expressions are available in many papers (e.g. [22]).

Ideas described above are apply for code implementation of Preisach model using Matlab. Magnetization is compute, using equation (2):

$$M = k\frac{M_S}{2}\left(\iint\limits_{s+} p(a,b)dadb - \iint\limits_{S-} p(a,b)dadb \right) \tag{2}$$

In equation (2), k represents a correction factor. Its identification is considering the equation of probability density function [23]. This factor computes iteratively with equation (3):

$$k = \frac{N}{D} \tag{3}$$

In equation (3), where N represents the number of measured points and D represents number of points for decreasing the resolution. Resolution is defined by the number of points used in graphic representation. The user has the possibility to set number of points used for representation. After each measured point, optimal values are recalculated. When no optimal values are found, the implemented algorithm uses the next point from measured values. It eliminates that measured point from calculations (decreases the initial imposed resolution). This parameter is computed in order to assure time efficiency between measured and calculated values of magnetization.

For calculation of integrals from equation (2), the following formula is used:

$$\iint\limits_{S+/S-} p(a,b)dadb = \sum_{\substack{k=1:N \\ j=1:N}} p(a_i, b_j) \cdot S_{cell} \tag{4}$$

In equation (4), S_{cell} represents the area of the cell in which Preisach triangle was mesh; a_i, b_j represent centers of cell in which Preisach triangle is mesh; p (a_i, b_j) represents probability density function numerically calculated in center of cell. Preisach triangle represents an isosceles right triangle in which ideas presented above are used. *a* and *b* represent the particular system of axis in which measured values are used according to Preisach model ideas. In this system, the state line is use for separation of positive and negative hysterons. In Figure 2 an example for 4 measured points, for a positive applied field is offer. The discretization is uniform.

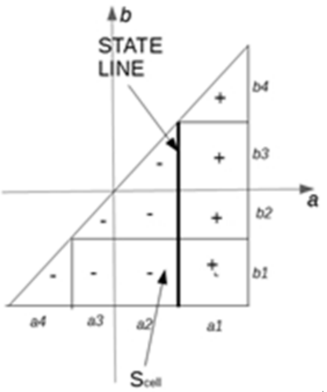

Figure 2. *Example of Preisach triangle meshing (N=4)*

In Figure 2 *a, b* represent in this case the center of a discretized cell, as shown for the case of *a1* and *b1*. In above example the number of cells is equal to 4 and the discretization is uniform, which is choosing in most cases.

Main problem in hysteresis models is choosing a probability density function that can characterize the tested sample. Numerous analytically expressions are available in literature (e.g. [24]).

The ideas presented above were had been implemented in Matlab. In this case, for probability density function a factorized Lorentzian distribution is use as suggested in [25]. The implemented probability density function is:

$$p(\alpha,\beta) = \frac{1}{[1+(\frac{\alpha - H_0}{\sigma H_0})^2][1+(\frac{\beta + H_0}{\sigma H_0})^2]} \tag{5}$$

In equation (5), α, β represent the tipping thresholds in Preisach model and H_0, σ represent unknown parameters that must be determined.

A mean square error between calculated (modeled) magnetization and measured (experimental) magnetization is use as a measure in determination of parameters. Error is determined using equation (6):

$$\varepsilon = \frac{1}{N}\sqrt{\sum_{1}^{N}\frac{(M_{mod} - M_{exp})^2}{(M_{exp})^2}} \tag{6}$$

In equation (6), M_{mod} represents the modeled (calculated) magnetization, M_{exp} represents the experimental (measured) magnetization, and N represents the number of measured values.

The main contribution of this paper is the implementation of a search algorithm for optimal values determination in case of two unknown parameters (H_0, σ) taking into account the applied field history. The proposed search algorithm is follows the next ideas. The user can input the values of these parameters: the number of general searches; the number of depth (refining) searches, for a chosen interval; a search range for each parameter and the initial values of the unknowns. Searches are around the indicated range, for each unknown, because some optimal solutions are supposed to be very close outside the indicated range. After the area of optimum values found, then refining searches made in iteration so that a minimum error is obtain. Search interval is influence by determined error and in case the code cannot produce any better results in that numerical area then the search stops. The code computes automatically the search step and it is equal with ratio between search interval for each parameter and number of refining searches.

Modification of search interval for each parameter of found error as in equation (7):

$$\mathrm{Sim} = \sqrt{\varepsilon} \tag{7}$$

In equation (7), *Sim* represents the modification of search interval after every iteration and ε represents mean square error determined in equation (6). Actualization of search interval is with next command from implemented code, at every step, as a product between previous found value and range limit values in which new search is perform.

The code also includes a function in which the user can set resolution at which search is made. This means that user can set number of points that is use in identification. If more points are available in identification, the time for results to be plotted also raises. In order to assure the fact that found values are local points, the code also includes a function that is use to do a sizing between experimental and calculated values. This function assures that the same tendency exists between experimental and determined values. When optimal values are discovered, around the found solutions search is at next iteration. The search stops after the number of maximum iterations imposed by user is attained. In addition, optimum obtained values are saving. Correlation of all these parameters needed efficient computation time and some implemented functions in Matlab where use.

For example, let us suppose that first value of estimated H_0 is 0 and search interval is equal to 10. Search interval is form from two equal length closed intervals centered in estimated value. Therefore, in this example, the range is [-5, 5]. Let us say that number of depth iteration is 5. The algorithm searches for first parameter in {-5, -2.5, 0, 2.5, 5}. For second unknown, let us suppose that estimated value is 0 and search interval is equal with 6. So the range is [-3, 3], determined by 6/2 at an equal distance from 0. For each first unknown, let us say H_0=-5, it calculates with σ=-3. It determines first error, ε1. For H_0=-5 it calculates with σ=0. It determines another error; let us name it ε2. And so on. It determines which error is smallest. In conclusion, it searches around optimum values found with a search interval, as a function of error.

Search is done recursively, as a function of imposed error and number of searches and refining searches. Parameters are searched iteratively, from the optimum value found at a certain time (local/global extreme value) at which the computed search interval is added/decreased. The search stops when the minimum obtained error is smaller or equal to the imposed one. Used search functions are presented below. A list of implemented functions and explications regarding their utility is offer in table 1.

Table 1. *List of implemented functions*

Function	Effect
Display.m	Displays Preisach triangle This function was only created in order to debug the code easier
Mesh.m	Divides Preisach triangle according to string values of H
InverseMeshing.m	Divides “inverse” of Preisach triangle according to values of magnetic field
EmptyTriangleGeneration.m	Generates Preisach triangle, triangle originally filled only with 0
InitialTriangleGeneration.m	Generates the initial Preisach triangle
PreisachTriangleGeneration.m	Generates Preisach triangle and: -initializes the unknown search parameters -calculates and displays the new magnetization values
Sizing.m	Makes the search algorithm to do searching for shape and size of the hysteresis cycle
StateLine.m	Implements the algorithm for the status line in Preisach triangle
Magnetization.m	Calculates magnetization like in Preisach classic model
DecreaseResolution.m	Used for an optimal calculation of unknown parameters taking into account the resolution
Parameters.m	Used parameters are transmitted through the definition of a class
RemoveCorner.m	Deletes the triangle of the square in which memorization is made Preisach model was saved

4. Results

A vibrating sample magnetometer, VSM 7304 produced by Lake Shore, was used in measurements. The sample (a magnetic subway card) was demagnetizing in initial step. A round sample was cut and use in experiments in order to know exactly the demagnetization factor. Acquisition control and the data processing are with included software. The measurements are saving in a text format. After the deletion of wrong measured points, the data were dividing by the saturation values as the active magnetic layer of material used in test could not be measure.

The experimental data were for three numerical tests, to verify the identification algorithm efficiency. For the first test, a portion from first quadrant of the major hysteresis cycle was use. The entire descending curve from the same hysteresis cycle was use for the second test. In the third test, simulations of the major hysteresis cycle are available. These tests were made in order to test the validity of implemented algorithm. The obtained results, with implementation of ideas described above are in Table 2.

Table 2. *Numerical results for a subway card*

Test type	Alpha (H_0)	Beta (σ)	Scaling	Time (s)	Resolution	General *m*in. square error [%]
0 degrees, quadrant 1	0.3	-0.45	0.084	67	20	0.47
0 degrees, descending curve	0.175	-0.2771	0.014	133	30	1.78
0 degrees, major cycle	0.195	-0.2602	0.016	267	28	3.26
90 degrees, descending curve	-0.05	-1	0.092	137	26	3.96
90 degrees, major cycle	0.975	-1	0.177	277	20	9.97

From table 2, in case of fewer points used in identification it can be observe that time is minim (e.g. 0 degrees when curve from quadrant 1 is used). Also, from table 2, the time needed for 1 point to use in calculation is approximate equal with 0.82 seconds. In case of a transversal applied field for quadrant 1, modeled and experimental results are show in Figure 3.

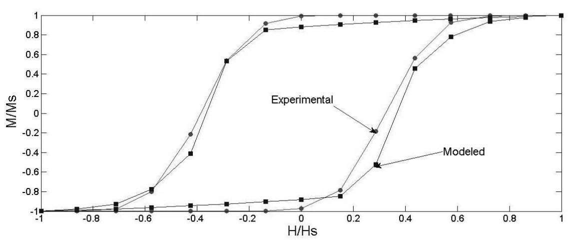

Figure 3. *Modeled and experimental results in case of transversal applied field*

For the case in which entire demagnetization curve of major hysteresis cycle is use, simulated graphic of experimental (measured) values and modeled (computed) values is presented in Figure 4.

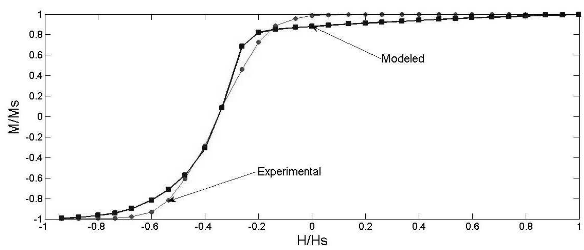

Figure 4. *Modeled and experimental curves for a longitudinal applied field*

It can be observe that only certain points were use in the 2 tests. The code computes the points in order to obtain faster optimum values of implemented probability density function. The way in which was modified was explained above. The precision of algorithm is influence by this parameter. If a better precision is desire, the price the user must pay is a longer waiting time.

In the same case of a longitudinal applied field, identification for a major hysteresis cycle is also available. Identification result of unknown parameters for probability density function is present in Figure 5.

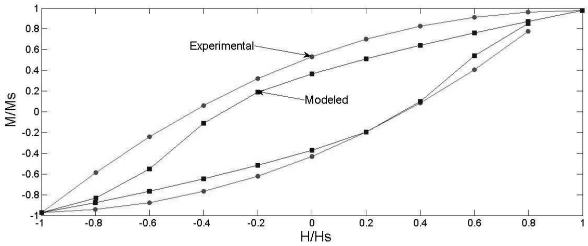

Fig 5. *Experimental and modeled curves for the major hysteresis cycle when longitudinal applied field is applied*

The number of measured values influences identification results. In case of a smaller number of used points in identification, as in Figure 5, the identification result between experimental and modeled curves is not so good. In case of a bigger number of points as in Figure 4, the fit is better. In addition, the general mean square error is bigger in case of major hysteresis cycle identification.

In order to test validity of implemented code for a dimensional hysteresis model tests are also on a transversal applied field. Simulation result when demagnetization curve is use in measurements is show in Figure 6.

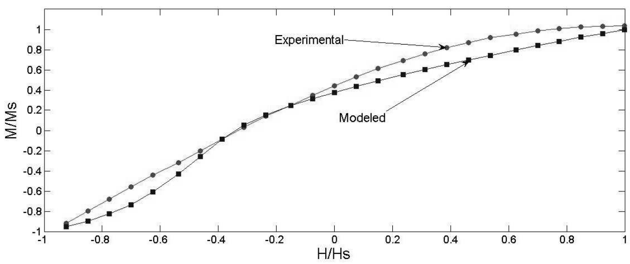

Figure 6. *Experimental and modeled curve for a 90 degrees applied field*

Simulation result for a transversal field shows the fit of the 2 curves in this case. Although the tests are for another direction of applied field, simulations in this case are good. The general mean square error is not so different from case of major hysteresis cycle in case of longitudinal applied field.

Identification is also for an experimental measured hysteresis cycle in same conditions as in previous case. Results are plot in Figure 7.

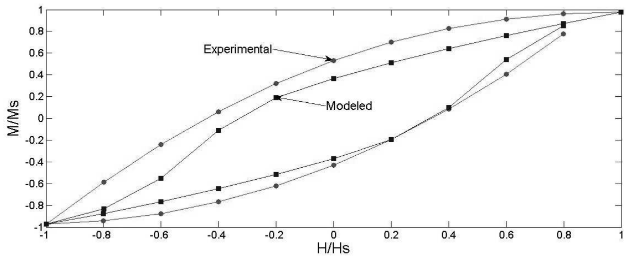

Figure 7. *Modeled and experimental curves in case of major hysteresis cycle for transversal applied field*

The algorithm, is not very good for approximating the entire hysteresis cycle (e.g. Figure 7). It is only suitable for some parts of the hysteresis cycle (e.g. demagnetization curve, initial magnetization curve). Also, the implementation of the Preisach hysteresis model in Matlab programming environment produces time advantages, because the code is optimized. Main disadvantage is that because it needs analytic formula, results are not so accurate. Future improvements regarding this direction must be done.

Results are similar to those presented in other papers (e.g. [26]). A qualitative comparison is difficult to make due to the specificity of probability density function for each material. Taking into account the novelty of the subject and the generality of implemented algorithm, we can say, from a quantitative point of view, that the values of identified parameters are in the same range as those for other soft magnetic materials.

This algorithm can be useful for identification of a Preisach vectorial hysteresis model. The search algorithm emphasizes the difficulty and multitude of decisions that must be done regarding influence of history. For this reason, one of the most frequent applications in which search applications are used, and this algorithm can be also used/extended is chess in which moves must be predicted as a function of opponent's future response. It was created to be used in search processes that involve finding optimum values for 2 variables. The code of recursive search functions is presented below.

```
%Search functions code implemented in Matlab
function result=RecursiveSimpleSearch (calculated_error, parameters, origin_alpha, origin_beta, search_interval, deep_searches, in_deeper_search)
  if(in_deeper_search<=0)
      result(1)=origin_alpha;
      result(2)=origin_beta;
      result(3)=calculated_error;
  else
      step = 2*search_interval/deep_searches;
      mimimum_error=calculated_error;
      alpha_optim=origin_alpha;
      beta_optim=origin_beta;
       for alpha_crt = (origin_alpha-search_interval) : step :
(origin_alpha+search_interval)
        for beta_crt = (origin_beta-search_interval) : step :
(origin_beta+search_interval)
              parameters.alpha=alpha_crt;
```

```
            parameters.beta=beta_crt;
        current_error=error_function_Preisach(parameters);
            if(current_error<minimum_error)
                minimum_error=current_error;
                alpha_optimum=alpha_crt;
                beta_optimum=beta_crt;
            end
        end
    end
     result=RecursiveSimpleSearch(minimum_error,
parameters, alpha_optim, beta_optim, step, deep_searches,
in_deeper_search-1);
  end
    function result=SimpleSearch(parameters,
search_interval, deep_searches, in_deeper_search)
result=SimpleSearchRecursiv(99999999999, parameters, 0,
0, search_interval, deep_searches, in_deeper_search);
```

It can be seen that this search is correlated with other functions, also described table 1.

5. Conclusions

The study and identification program developed allows obtaining useful results with hysteresis modeling of magnetic materials by proposing a solution for the efficient use of hysteresis models in the analysis of electromagnetic devices. The tests performed may serve to develop research on efficient modeling of vector hysteresis in electrical engineering.

In this paper a two variable search algorithm was implemented for determining optimal values in any case of probability density function representation taking into account a minimum set of measured data. This algorithm includes storage and updates of history for applied field and magnetization, defining property for hysteresis phenomenon.

This algorithm is faster than others are because only optimal measured points are plot in numerical representation. Precision of identification is good and consists in automatically adjustment of the control parameters of the search (mesh fineness of Preisach plane, the number of intervals required by search and error).

The Preisach hysteresis model was implement in Matlab programming environment using routines in order to assure an efficient time of time. The efficiency of program is influence by the input data set. The time needed in order for results to be determined is also available. This time is influence by the imposed precision and the probability density function chosen for simulation. The price that user must pay for a deeper precision is a longer waiting time. Because the implemented code consists in advanced programming it can be, say that implementation is one of the fastest for software that must take into account history of input. In addition, optimal determination of parameters for a factorized Lorentzian function is present. User can set precision. Taking into account, the results it was proved that the model could be useful in electrical engineering applications where time is very important.

This algorithm can be generalized for more variables. It can be used for numerical problems that involve identification of unknown parameters in problems that involve hysteresis. The tests can be extended for more materials, in order to obtain the optimum function for a certain material. Also the study can be used in efficient implementation of hysteresis modelling in computer aided design. A more detailed analysis of the report accuracy-numerical efficiency can be a future direction for studying hysteresis modelling.

The possibility of extending this research is very big taking into account the novelty of the treated subject. A generalization of the algorithm for other types of materials is a future subject.

Acknowledgements

The work has been funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labour, Family and Social Protection through the Financial Agreement POSDRU/107/1.5/S/76909.

References

[1] I.D.Mayergoyz, Mathematical Models of Hysteresis, Springer Verlag, New York, 1990

[2] I. D. Mayergoyz, G. Bertotti, The science of hysteresis, Academic Press, London, 2005

[3] G. Bertotti, Hysteresis in Magnetism, Academic Press, London, 1998

[4] E.Della Torre, Magnetic Hysteresis, IEEE Press, Piscataway, 1999

[5] A. Visintin, Differential Models of Hysteresis, Springer, Berlin, 1994

[6] A. Ivanyi, Hysteresis Models in Electromagnetic Computation, Akademiai Kiado, Budapest, 1997

[7] [7] F. Preisach, “Uber die magnetische”, Nachwirkung, Zeitschrift fur Physik, No. 94, 1935, pp. 277-302

[8] E. C. Stoner, F. R. S. , E. P. Wohlfarth, “A mechanism of magnetic hysteresis in heterogeneous alloys”, IEEE Trans. on Magn., vol. 27, no. 4, 1991, pp. 3475-3518

[9] L. Dupre, J. Melkebeek, “Electromagnetic hysteresis modelling: from material science to finite element analysis of devices”, ICS-Newsletter, vol. 10, no. 3, 2003, pp. 3-14

[10] E. Cardelli, E. D. Torre, A. Faba, “Phenomenological modeling of magnetic hysteresis”, ICS-Newsletter, vol. 17, no. 1, 2010, pp. 3-17

[11] D. L. Atherton, J. R. Beattie, "A mean field Stoner-Wohlfarth hysteresis model", IEEE Trans. on Magn., vol. 26, no. 6, 1990,pp. 3059-3063

[12] E. Della Torre, G. Kádár, "Vector Preisach and the moving model", J. Appl. Phys., vol. 63, no. 8, 1988, pp. 3004-3006

[13] E. D. Torre, F. Vajda, “Parameter identification of the complete-moving-hysteresis model using major loop data”, IEEE Trans. on Magn., vol. 30, no. 12, 1995, pp. 4987-5000

[14] J. Fuzi, “Analytical approximation of Preisach distribution functions”, IEEE Trans. On Magn., vol. 39, no. 5, 2003, 1357-1360

[15] B. Van de Wiele, L. Vandenbossche, L. Dupre, D. Daniel Zutter, “Energy considerations in a micromagnetic hysteresis model and the Preisach model”, J. Appl. Phys, vol. 108, no. 10, 2010, pp. 103902 - 103902-10

[16] S. Motoasca, G. Scutaru, “Hysteresis modelling of soft magnetic materials using LabVIEW programs”, Advances in Electrical and Computer Engineering, vol. 10, no. 2, 2010, pp. 94-97

[17] A. Stancu, L. Stoleriu, P. Andrei, “Vectorial Preisach type model designed for parallel computing”, J. of Magnetism and Magnetic Mat., vol. 316, no. 2, 2007, pp. 309-312

[18] G. Radons, F. Hebe, R. Lange, S. Schubert, “On the dynamics of non-linear hysteretic systems”, in press

[19] M. Kuczman, “Vector Preisach hysteresis modelling: Measurement, identification and application”, Physica B, No. 406, 2011, pp. 1403-1409

[20] A. A. Adly, “Numerical implementation and testing of new vector isotropic Preisach-type models”, IEEE Trans. On Magn., vol. 30, no, 6, 1994, pp. 4383-4385

[21] S. Chapra, „Applied numerical methods with Matlab for Engineers and Scientists 3rd edition”, McGraw-Hill Higher Education, 2012

[22] S. M. Sina et al., “Preisach analysis of sputtered SmCo thick films”, Journal of Applied Physics, vol. 113, no. 14, 2013, pp. 143901-143901-4

[23] O. Tabara, Vectorial hysteresis models in magnetism, PhD thesis, University “Politehnica” of Bucharest, 2013

[24] J. Fuzi, “Analytical approximations of Preisach distribution function”, IEEE Trans. on Magn., vol. 39, no. 3, 2003, pp. 1357-1360

[25] O. Henze, W. M. Rucker, “Identification procedures of Preisach model”, IEEE Trans. On Magn., vol. 36, no. 2, 2002, pp. 833-836

[26] X. Tan, J. S. Baras “Adaptive identification and control of hysteresis in smart materials”, IEEE Trans. on Automatic Control, vol. 50, no. 6, 2005, pp. 827-839