Parametric Geometric Metamodel of Nonlinear Magnetostatic Problem Based on POD and RBF Approaches

Allaa Eddine Boumesbah, Thomas Henneron

Parametric Geometric Metamodel of Nonlinear Magnetostatic Problem Based on POD and RBF Approaches

Allaa Eddine Boumesbah and Thomas Henneron

Univ. Lille, Arts et Metiers Institute of Technology, Centrale Lille, Yncrea Hauts-deFrance, ULR 2697 - L2EP, F -59000 Lille, France.

A parametric geometric metamodel is built for a nonlinear magnetostatic problem, using proper orthogonal decomposition approach combined with radial basis functions interpolation. Furthermore, the geometrical variation of the problem is modeled using a RBF interpolation for smooth mesh deformation. The metamodel is applied for a single-phase EI inductance, the aim is to create precise flux cartographies based on few solutions of the original finite element (FE) model. The results show that the POD-RBF approach can reduce efficiently the evaluation time of a parametric nonlinear magnetostatic problem.

***Index Terms*— Model Order Reduction, Proper Orthogonal Decomposition, Radial Basis Function, mesh deformation.**

I. Introduction

The finite element (FE) method is widely used to model electromagnetic devices. This method ensures precise results at the expense of a consequent computation time, especially when dealing with parametric problems. Then, model order reduction (MOR) methods had proven their efficiency in reducing the computation time for parametric studies. The proper orthogonal decomposition (POD) is one of the most popular MOR approaches [1]. Based on the solutions of the FE model for different values of parameters (called snapshots), the POD approximates the solution of the FE model in a reduced basis. Then, the parametric FE model is projected onto a reduced basis, decreasing the order of the numerical model to be solved for new values of parameters. Another approach consists in constructing a metamodel by interpolating the parametric FE solution expressed in a reduced basis for new values of parameters without solving the matrix system and avoiding the nonlinear iterative scheme. Based on the principle, different approaches had been proposed in the literature, such as the POD with interpolation (PODI) [2], using an optimization process, or the orthogonal interpolation method (OIM) [3], using polynomial interpolations, for example. Another approach has also been proposed combining the POD with the radial basis function (RBF) interpolation. Initially developed for mechanical and thermal problems [4] [5], it had been also applied to a nonlinear magnetostatic problem [6]. Moreover, when the problem involves geometric parameters, an adaption of the FE mesh is required. Remeshing the geometry at each modification can deteriorate the precision of the FE model by introducing significant numerical noise. A solution to preserve the problem's consistency is to deform the initial mesh using the RBF interpolation [7][8]. The method consists of imposing a displacement on a set of nodes and interpolating the remaining nodes' position.

In this work, we propose to build a metamodel of a nonlinear magnetostatic parametric problem involving geometric and electric parameters using the POD-RBF approach to approximate the FE solution and the RBF interpolation for the mesh deformation. First, the study case and the FE model are represented in sections II and III. The RBF interpolation method is introduced in section IV. In section V, results for mesh deformation using the RBF interpolation are presented. The approach to build the POD-RBF metamodels is explained in section VI, and the results are presented in section VII.

II. Study Case

The study case is a 2D single-phase EI inductance, with one electrical parameter and two geometrical parameters. An initial geometry is meshed with 36859 elements and 18553 nodes (Fig. 1.). The geometric parameters are the thickness of the air gap e varying from $0.1\ mm$ to $0.9\ mm$, the width of the central column of the magnetic core d varying from $20\ mm$ to $40\ mm$. The electric parameter is the phase current i of the winding defined in the interval $[0{,}50]\ A$.

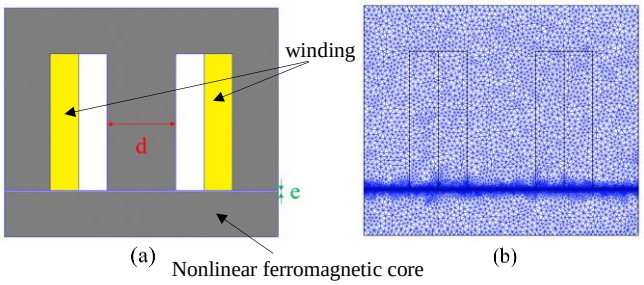


Fig. 1. The single-phase EI inductance (a) and associated mesh ($d = 30\ mm$ and $e = 0.5\ mm$) (b).

III. Finite Element Model

The governing equations for the nonlinear magnetostatic FE model are derived from Maxwell's equations. The vector potential formulation used to solve the problem is given by:

$$\boldsymbol{curl}\big(\nu(\boldsymbol{B})\ \boldsymbol{curl}(\boldsymbol{A})\big) = \boldsymbol{J_s} \tag{1}$$

where ν is the reluctivity depending on the magnetic flux density $\boldsymbol{B}$, $\boldsymbol{A}$ the magnetic vector potential and $\boldsymbol{J_s}$ the source current density depending on the current i. The discretized nonlinear 2D FE equation system can be expressed by:

$$\boldsymbol{S}(\boldsymbol{X}).\boldsymbol{X} = \boldsymbol{F} \tag{2}$$

where $\boldsymbol{X} \in \boldsymbol{R}^n$ is the vector of unknowns corresponding to the components of the magnetic vector potential, $\boldsymbol{S}(\boldsymbol{X})_{n\times n}$ the curl-curl stiffness matrix and $\boldsymbol{F} \in \boldsymbol{R}^n$ the source vector. Then, the FE solution $\boldsymbol{X}$ depends on the current i, the width of the central column d and thickness of the air gap e.

IV. Radial Basis Function Interpolation Method

The RBF interpolation consists in approximating a multivariate function $f(\boldsymbol{x})$, using a linear combination of basis functions ϕ, pre-calculated for N_t training points. The approximation of $f(\boldsymbol{x})$ is given by:

$$f(\boldsymbol{x}) \approx \sum_{i=1}^{N_t} \alpha_i \phi_i(\boldsymbol{x}) \quad (3)$$

Where $\phi_i(\boldsymbol{x}) = \phi(\|\boldsymbol{x} - \boldsymbol{x}_i\|)$ is a radial function depending on the Euclidean distance between $\boldsymbol{x}$ and $\boldsymbol{x}_i$; and α_i is its associated coefficient. Different kinds of functions can be used to define $\phi_i(\boldsymbol{x})$ such as Gaussian functions for example. The determination of the coefficients α_i can be done by solving the matrix system:

$$\boldsymbol{Y} = \boldsymbol{G}\boldsymbol{\alpha} \quad (4)$$

with $\boldsymbol{Y} = \left[f(\boldsymbol{x}_1), \dots, f(\boldsymbol{x}_{N_t})\right]^t$, $\boldsymbol{\alpha} = \left[\alpha_1, \dots, \alpha_{N_t}\right]^t$

and $\boldsymbol{G} = \begin{bmatrix} \phi_1(\boldsymbol{x}_1) & \cdots & \phi_1(\boldsymbol{x}_{N_t}) \\ \vdots & \ddots & \vdots \\ \phi_{N_t}(\boldsymbol{x}_1) & \cdots & \phi_{N_t}(\boldsymbol{x}_{N_t}) \end{bmatrix}$

This approach can be generalized for vector function $\boldsymbol{f}(\boldsymbol{x}) \in \boldsymbol{R}^q$ by interpolating each component.

For a new set of inputs $\boldsymbol{x}_{new}$, $f(\boldsymbol{x}_{new})$ can be approximated by:

$$f^j(\boldsymbol{x}_{new}) = \sum_{i=1}^{N_t} \alpha_i^j \phi_i(\boldsymbol{x}_{new}) \quad (5)$$

Where f^j is the j^{th} component of the function f and α^j is its corresponding coefficients.

V. Mesh Deformation Based on RBF Interpolation

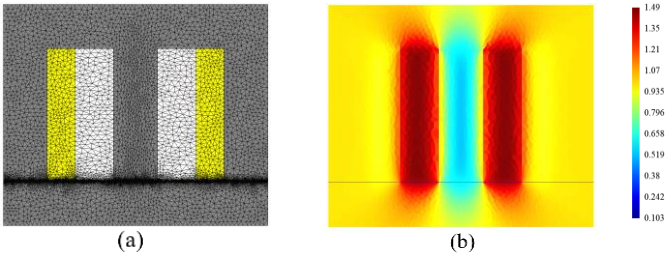
In order to model the geometric variation of the problem, mesh deformation is performed using the RBF interpolation [7][8] for two consecutive times to consider the two varying geometric parameters e and d. For each mesh deformation, two sets of nodes are defined, N_f nodes where the displacements are imposed and N_i nodes where the displacements are approximated. In this case, the function to be interpolated is the displacement function denoted $\boldsymbol{d}(\boldsymbol{x})$. By applying the RBF interpolation, the displacement $\boldsymbol{d}(\boldsymbol{x}_k)$ for a node $n_k \in N_i$ is approximated by:

$$\boldsymbol{d}(\boldsymbol{x}_k) = \sum_{i=1}^{N_f} \alpha_i \phi_i(\boldsymbol{x}_k) \quad (6)$$

An indicator for the conformity of the mesh d_e is defined by:

$$d_e = \frac{det_{e\_def}}{det_{e\_init}} \quad (7)$$

with det_{e_def} and det_{e_init} are the determinants of the deformed and the initial element e, respectively. The indicator is positive in the case of a conformal mesh, a negative indicator indicates an overlapping between the elements, and thus the non-conformity of the deformed mesh. Figure 2 presents an example of mesh deformation.



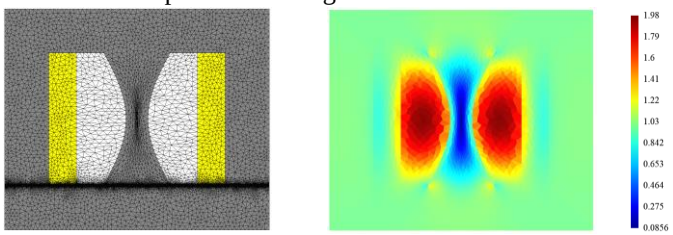
(a) (b)

Fig. 2. Deformed geometry (a) and the distribution of the deformation indicator (b), for $d = 20\ mm$ and $e = 0.1\ m$.

The RBF interpolation can also be used to model a more complex mesh deformation; another example is to impose a sinusoidal displacement d_x on the nodes on the border of the central column of the magnetic core, such as:

$$d_x = \pm 10 \sin((20.5 - y)\pi/60) \quad (8)$$

The deformed mesh and the distribution of the deformation indicator are represented in Fig. 3.



(a) (b)

Fig. 3. Deformed geometry (a) and the distribution of the deformation indicator (b), for a sinusoidal displacement.

VI. Metamodel Based on POD-RBF

A. Principle of POD combined with RBF

The POD-RBF approach is applied to the parametric FE model [4][5][6]. The method consists of first gathering a set of data, using the method of snapshots, by solving the original FE model (2) for a set of parameter values:

$$\boldsymbol{M_s} = [\boldsymbol{X}_1 \quad \cdots \quad \boldsymbol{X}_m] \quad (9)$$

Where $\boldsymbol{M_s}$ is the snapshots matrix, m is the number of snapshots and $\boldsymbol{X}_i$ is the FE solution at the i^{th} set of parameter values. Then, the snapshots matrix is decomposed using the singular value decomposition (SVD).

$$\boldsymbol{M_s} = \boldsymbol{Y}\boldsymbol{\Sigma}\boldsymbol{W}^t \quad (10)$$

The left orthogonal matrix of the SVD decomposition $\boldsymbol{Y}$ corresponds to the reduced basis, and the right orthogonal matrix $\boldsymbol{W}$ is composed of the FE solution projected into the reduced basis for each set of parameter values. The POD reduced basis $\boldsymbol{\Psi}$ of size $n \times m$, is given by:

$$\boldsymbol{\Psi} = \boldsymbol{Y}\boldsymbol{\Sigma} \quad (11)$$

For any new set $\boldsymbol{x}_{new}$ of parameter values, the approximated solution $\boldsymbol{X}^{ap}$, can then be defined by:

$$\boldsymbol{X}^{ap} = \boldsymbol{\Psi}\boldsymbol{V}^t(\boldsymbol{x}_{new}) \quad (12)$$

Where $\boldsymbol{V}^t(\boldsymbol{x}_{new})$ is obtained by performing a RBF interpolation on the right orthogonal matrix $\boldsymbol{W}^t$, each element v_l for $l = 1, \dots, m$ is interpolated for the new set $\boldsymbol{x}_{new}$ by:

$$v_l(\boldsymbol{x}_{new}) = \sum_{i=1}^{m} \alpha_{li} \phi_i(\boldsymbol{x}_{new}) \quad (13)$$

The coefficients α_{li} are determined by solving an equation system as in (4).

B. Greedy algorithm

The choice of snapshots has an impact on the precision and the efficiency of the POD-RBF approach, the aim is to build a precise metamodel using a low number of snapshots. Two greedy algorithms are proposed, one based on a precomputed snapshots library [6], where the exact error of the metamodel can be computed, and the other based on the evaluation of the residual of (2) with approximated solution from (12) to obtain

an error approximation without having to solve the FE model for a large number of parameter values sets. We denote $\boldsymbol{p}$ a coordinate in the parameter space such as $\boldsymbol{p} = (i, e, d)$.

The first algorithm consists of; first, calculating a sufficiently large snapshots library $\boldsymbol{M}_X$, for a set of parameters chosen based on the knowledge of the engineer of each studied device . Then, an iterative process is initiated, where at each iteration a new snapshot corresponding to the maximum relative error of coordinate $\boldsymbol{p}_m$ is considered to update the POD-RBF metamodel. The metamodel and the error are evaluated in the grid of parameters space used to precompute the snapshots library. The process is repeated until the stop criterion is satisfied.

Algorithm 1 Greedy algorithm with exact error evaluation

Input: $\boldsymbol{M_X} = [\boldsymbol{X}_1 \quad \cdots \quad \boldsymbol{X}_Q]$, $\boldsymbol{p}_1$
Output: $\boldsymbol{\Psi}$, α_{li}
- $m = 0$
while $max(\boldsymbol{e}) > \epsilon_1$ **do**
- $m = m + 1$
if $m > 1$ **then**
- select the coordinate corresponding to the maximum error such as $\boldsymbol{p}_m = argmax(\boldsymbol{e})$
end if
- $\boldsymbol{p} \leftarrow [\boldsymbol{p}, \boldsymbol{p}_m]$
- $\boldsymbol{M_s} \leftarrow [\boldsymbol{M_s}, \ \boldsymbol{X}_{\boldsymbol{p}_m}]$
- perform a SVD on $\boldsymbol{M_s}$, update $\boldsymbol{\Psi}$ and α_{li}
- evaluate the error $\boldsymbol{e} = [e_k]_{k=1}^{Q}$ with $e_k = \frac{\|\boldsymbol{X}_k - \boldsymbol{X}_k^{ap}\|_2}{\|\boldsymbol{X}_k\|_2}$ and $\boldsymbol{X}_k^{ap}$ computed by (12)
end while

For the second algorithm, the metamodel and the norm of residual $\boldsymbol{r}$ are evaluated on a sufficiently large grid of parameters space $\boldsymbol{\Omega}$, defined by the user. At each iteration a new snapshot is added for the coordinate $\boldsymbol{p}_m$ corresponding to the greatest norm of the residual. The process is repeated until the stop criterion set on the residual norm is met.

Algorithm 2 Greedy algorithm with residual evaluation

Input: $\boldsymbol{p}_1$, $\boldsymbol{\Omega} = [\omega_1, \ldots, \omega_Q]$
Output: $\boldsymbol{\Psi}$, α_{li}
- $m = 0$
while $max(\boldsymbol{r}) > \epsilon_2$ **do**
- $m = m + 1$
if $m > 1$ **then**
- select the coordinate corresponding to the greatest residual such as $\boldsymbol{p}_m = argmax(\boldsymbol{r})$
end if
- $\boldsymbol{p} \leftarrow [\boldsymbol{p}, \boldsymbol{p}_m]$
- $\boldsymbol{M_s} \leftarrow [\boldsymbol{M_s}, \ \boldsymbol{X}_{\boldsymbol{p}_m}]$
- perform a SVD on $\boldsymbol{M_s}$, update $\boldsymbol{\Psi}$ and α_{li}
- evaluate the residual $\boldsymbol{r} = [r_k]_{k=1}^{Q}$ with $r_k = \left\|\boldsymbol{F} - \boldsymbol{S}(\boldsymbol{X}_k^{ap})\boldsymbol{X}_k^{ap}\right\|_2$ and $\boldsymbol{X}_k^{ap}$ computed by (12)
end while

The stop criterion for the first algorithm ϵ_1 is set to a maximum error between the FE solution and the metamodel solution lower than 5%. While for the second algorithm, the stop criterion ϵ_2 is set to a maximum norm of the residual lower than 5. It is important that the snapshots library $\boldsymbol{M}_X$ for the first algorithm, and the grid $\boldsymbol{\Omega}$ for the second algorithm, are large enough. Because, in the first case, the snapshots are selected from the library, and in the second, the snapshots are chosen within the coordinates of the grid. Having a reduced library or a coarse grid will reduce the precision of the metamodel.

VII. Results

A. FE model

First, the original FE model in (2) is solved for a regular grid of 2574 points, for 9 air gap thicknesses e from 0.1 mm to 0.9 mm, 11 d values from 20 mm to 40 mm and 26 current values i from 0 to 50 A. Magnetic flux associated with the winding is calculated as a function of the three varying parameters. Cartographies of linkage flux as a function of d and current i, for fixed values of e, are represented in Fig. 4.

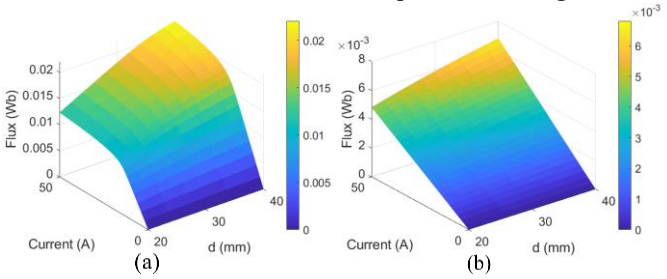


Fig. 4. Original FE model flux cartographies as a function of d and current i for $e = 0.1\ mm$ (a) and $e = 0.9\ mm$ (b).

The required time to solve the FE model for the 2574 calculation points is 16 min, using an i7 CPU with a 16 Go RAM. The results of the original FE model are taken as a reference. Fig. 5 presents the distribution of the magnitude of the magnetic flux density $\boldsymbol{B}$.

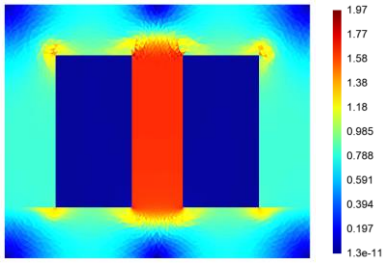


Fig. 5. Magnitude distribution of $\boldsymbol{B}(T)$ for $e = 0.1\ mm$, $d = 0.9\ mm$ and $i = 40A$.

B. POD-RBF metamodel using algorithm 1

A POD-RBF-based metamodel is built using the first greedy algorithm. The snapshots library $\boldsymbol{M_X}$ is composed of the FE model solutions for the 2574 defined calculation points. To satisfy the stop criterion, the greedy algorithm selects 151 snapshots. Once the snapshots library is calculated, the total time required for the greedy algorithm convergence and the evaluation of the POD-RBF metamodel for the 2574 calculation points, is 4 min, an additional 16 min is required to create the snapshots library. To assess the precision of the metamodel, the

error between the fluxes calculated by the metamodel $\boldsymbol{\Phi}^{ap}$ and the FE model $\boldsymbol{\Phi}^{FE}$, is defined as:

$$\boldsymbol{e}^{\Phi} = \frac{|\boldsymbol{\Phi}^{FE} - \boldsymbol{\Phi}^{ap}|}{|\boldsymbol{\Phi}^{FE}|} \times 100 \quad (17)$$

The error distributions as a function of d and i for fixed e values are presented in Fig. 6.

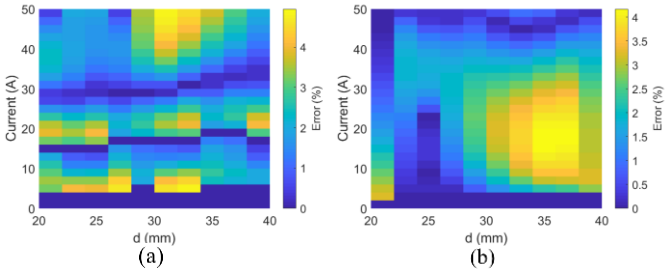


Fig. 6. First algorithm POD-RBF metamodel error distribution for $e = 0.1\ mm$ (a) and $e = 0.9\ mm$ (b).

The global mean relative error between the flux cartographies obtained using the metamodel and the original FE model is 1.6%, while the maximum local error is 5%. The first algorithm chooses the snapshots for the coordinates corresponding to the maximum error, thus the maximal local error is relatively low and the error distribution is relatively even. Fig. 7 presents the difference of the magnitude distributions of $\boldsymbol{B}$ obtained from the FE model and the POD-RBF approximation for the same parameter values as in Fig. 5. The magnitudes of the error are small compared with those of the magnetic flux density (Fig. 5). The maximal values of the error are located in the internal corners where the ferromagnetic core is saturated.

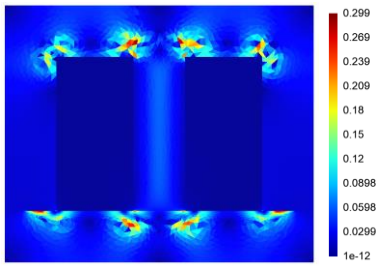


Fig. 7. Difference of the magnitude distribution of $\boldsymbol{B}(T)$ for $e = 0.1\ mm$, $d = 0.9\ mm$ and $i = 40A$.

C. POD-RBF metamodel using algorithm 2

A second POD-RBF metamodel is built, using the residual-based greedy algorithm. The metamodel is evaluated on a grid composed of the same 2574 points where the FE model is solved. 206 snapshots are required to reach the stop criterion. The convergence of the residual-based algorithm requires 3 h, it is significantly higher than the time required by the exact error-based algorithm, as in the second algorithm the snapshots are not precomputed and thus the FE model must be solved at each iteration to get a new snapshot. Furthermore, the evaluation of the residual generates an additional computational time, as it requires the assembly of the FE matrix for every point of the grid and at each iteration. The convergence time of the residual-based algorithm is also higher than the required time to solve the FE model on the defined grid. Indeed, solving equation (2) consumes more time than matrix assembly; however, in the iterative process of the residual-based algorithm, the matrix assembly step is performed over 500000 times, which explains the increase of calculation time. The distributions of the error on the flux cartographies obtained using this metamodel are represented in Fig. 8.

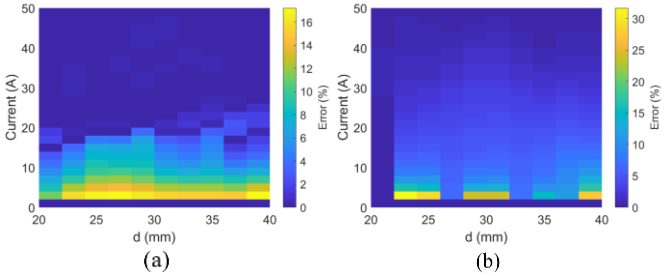


Fig. 8. Second algorithm POD-RBF metamodel error distribution for $e = 0.1\ mm$ (a) and $e = 0.9\ mm$ (b).

The global mean error on the flux calculated using the residual-based metamodel is 4.1%, while the maximum local error can reach 51%. As shown in Fig. 8., the error is mainly concentrated around the low values of current. The residual used as the criterion of snapshots choice in this algorithm is an absolute value; thus, the algorithm often neglects the points where the solution is low, even if the relative error in these coordinates is high. The residual-based algorithm selected more snapshots than the exact error-based algorithm. However, the precision of the first algorithm is better. This is due to the fact the residual does not give a precise evaluation of the error; instead, it gives an approximation of the error without having to calculate the exact solution.

Fig. 9 presents the difference of the magnitude distribution of $\boldsymbol{B}$ obtained from the FE model and the residual-based metamodel for the same parameter values as in Fig. 5. The magnitudes of the error are of the same order as those in Fig. 7.

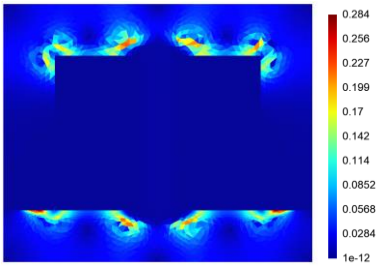


Fig. 9. Difference of the magnitude distribution of $\boldsymbol{B}(T)$ for $e = 0.1\ mm$, $d = 0.9\ mm$ and $i = 40A$.

D. Comparison

Two POD-RBF metamodels had been established, for a magnetostatic nonlinear parametric problem, allowing to approximate the FE solution as a function of three parameters. The two metamodels are built using two different algorithms, with different choice criteria for the snapshots. These criteria will impact the precision of the metamodels and the convergence of the algorithms. In order to evaluate the convergence rate of the two algorithms, two types of errors are defined, the maximum error in the sense of Euclidean norm e_{n2}, representing the maximum local error, and the mean error in the sense of Frobenius norm e_{Fro}, representing the global mean error.

$$e_{n2} = max\left(\frac{\|\boldsymbol{X}_i - \boldsymbol{X}_i^{ap}\|_2}{\|\boldsymbol{X}_i\|_2}\right)_{i=1,\dots,nc} \quad (14)$$

$$e_{Fro} = \frac{\|\boldsymbol{X} - \boldsymbol{X}^{ap}\|_{Fro}}{\|\boldsymbol{X}\|_{Fro}} \quad (15)$$

The evolution of error represented by the two norms as a function of the number of iterations, for the two algorithms is plotted in Fig. 10.

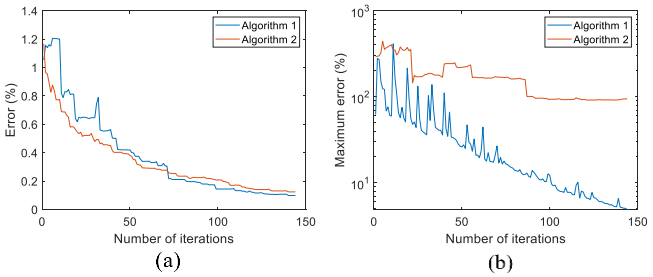


Fig. 10. Evolution of the error as a function of the number of iterations, in the sense of Frobenius norm (a) and Euclidean norm (b).

Although globally converging in the same way (Fig. 10.a), the maximum error converges best for the first algorithm. The first algorithm chooses at each iteration the snapshots corresponding to the higher maximum error, resulting in fast convergence of the maximum error. On the other hand, the residual-based algorithm chooses the snapshots corresponding to the higher residual norm; thus, if the maximum relative error does not represent an important residual norm, many iterations can pass without any improvement of the maximum relative error, which can be seen in the constant parts in Fig. 10.b. The impact of the choice criteria on the selected snapshots can be seen in Fig.11. The distribution of snapshots coordinates in the parameters space selected by the two algorithms is plotted for the 151 first snapshots.

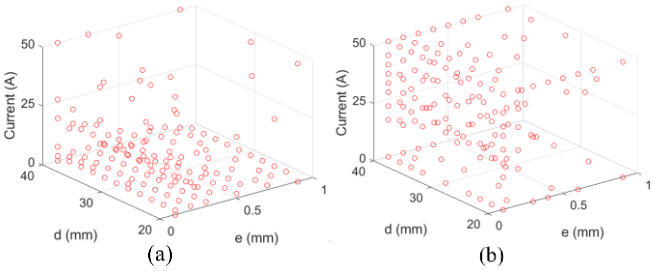


Fig. 11. Snapshots selected by the first (a), and the second algorithm (b).

The distribution of the snapshots coordinates confirms that the first algorithm favors the points where the solution is low and the relative error is high, these points correspond to the lowest current values (Fig. 11.a). Contrariwise, the residual-based algorithm favors the points where the solution and the absolute error are high, these points correspond to the higher current values (Fig. 11.b).

The performances of the metamodels resulting from the two algorithms, in terms of precision and computational time for the same number of snapshots, are summarized in Table 1.

The POD-RBF is an interpolation-based method; it offers the advantage of approximating the solution without accessing the FE full order matrix system. Nevertheless, the residual-based algorithm requires the assembly of the FE matrix, for each calculation point at each iteration, in an additional time-consuming step, in comparison with the first algorithm. Moreover, the residual only gives an approximation of the exact error. The snapshots selected by the residual-based algorithm do not necessarily correspond to the most important error; thus, more snapshots are required to reach the same precision as the first algorithm.

TABLE I
PERFORMANCES SUMMARY

Model	Total time (min)	Time per input (ms)	Mean error (%)	Maximum error (%)
Original FE	16	373	/	/
Algorithm 1	4+16	0.1	1.7	5
Algorithm 2	90	0.1	5.5	87

The evaluation time of a new set of parameter values for the POD-RBF-based metamodels is significantly lower than the solving time for the FE model. The necessary time to build a POD-RBF metamodel depends on the used algorithm. Nevertheless, omitting the construction time of the metamodel, the use of a POD-RBF-based metamodel can significantly reduce the computation time.

VIII. CONCLUSION

A parametric nonlinear magnetostatic problem has been studied. A consistency preserving mesh deformation is carried out using a RBF interpolation. Two POD-RBF metamodels of the FE solution are built, following two different algorithms. The type of algorithm impacts the choice of snapshots, the accuracy and the construction time of the metamodel. The established metamodels allow a significant reduction of the calculation time. POD-RBF metamodels of the parametric FE solution can be used in an optimization process for different applications or to be coupled with other numerical models to simulate a more complex system.

REFERENCES

[1] L. Sirovich, "Turbulence and the dynamics of coherent structures. I. Coherent structures," *Quarterly of applied mathematics*, vol. 45, no 3, pp. 561-571, 1987.

[2] R.R. Rama, S. Skatulla, C. Sansour, "Real-time modelling of the heart using the proper orthogonal decomposition with interpolation," *International Journal of Solids and Structures*, 2015.

[3] M. Farzam Far, F. Martin, A. Belahcen, L. Montier and T. Henneron, "Orthogonal interpolation method for order reduction of a synchronous machine model," *IEEE Transactions on Magnetics*, vol. 54, no. 2, pp. 1-6, Feb. 2018.

[4] W.P. Adamczyk, Z. Ostrowski and A. Ryfa, "Development of a non-destructive technique for measuring thermal conductivity of material with small anisotropy based on application of the reduced order technique," *Measurement*, vol. 165, p. 108078, 2020.

[5] V. Buljak, "Proper orthogonal decomposition and radial basis functions algorithm for diagnostic procedure based on inverse analysis," *FME Transactions*, vol. 38, no 3, p. 129-136, 2010.

[6] T. Henneron, A. Pierquin and S. Clénet, "Surrogate model based on the POD combined with the RBF interpolation of nonlinear magnetostatic FE model," *IEEE Transactions on Magnetics*, vol. 56, no. 1, pp. 1-4, 2020.

[7] A. de Boer, M.S. van der Schoot and H. Bijl, "Mesh deformation based on radial basis function interpolation," *Computers & structures*, vol. 85, no 11-14, p. 784-795, 2007.

[8] T. Henneron, A. Pierquin and S. Clénet, "Mesh Deformation Based on Radial Basis Function Interpolation Applied to Low-Frequency Electromagnetic Problem," *IEEE Transactions on Magnetics*, vol. 55, no. 6, pp. 1-4, 2019.