Separated representation of the finite element solution of nonlinear magnetostatic problem based on non-intrusive Proper Generalized Decomposition

Thomas Henneron, Stephane Clenet

Separated Representation of the Finite Element Solution of Nonlinear Magnetostatic Problem Based on Non-Intrusive Proper Generalized Decomposition

T. Henneron[a,*], S. Clénet[a]

[a] *Univ. Lille, Arts et Métiers Institute of Technology, Centrale Lille, Junia, ULR 2697 - L2EP, 59000, Lille, France*

Abstract

The Proper Generalized Decomposition has shown its efficiency to solve parameterized problems in nonlinear system events when it is combined with the Discrete Empirical Interpolation Method. Nevertheless, the solution of finite element model with the Proper Generalized Decomposition framework requires to have access to matrices and vectors of the discretized problem, which makes the method highly intrusive. In this context, based on a set of finite element solutions for a set of input parameters, a surrogate model can be developed applying a non-intrusive Proper Generalized Decomposition approach. The proposed non-intrusive approach is based on a canonical decomposition of the finite element solutions combined with an interpolation method. We then obtain a surrogate model approximating the finite element solutions for a wide range of parameters. The surrogate model, given its evaluation speed, can be used for real-time applications. In this paper, the proposed non-intrusive Proper Generalized Decomposition approach is employed to approximate a nonlinear magnetostatic problem and is applied to a single phase standard transformer and to a three-phase inductance.

Keywords:
Nonlinear magnetostatic problem, Model Order Reduction, Proper Generalized Decomposition.

1. Introduction

To study low frequency electromagnetic devices, the Finite Element (FE) method is widely used. The FE method enables accurate results but requires a long computational time due to numerical or physical features such as a high number of Degrees of Freedom (DoF) in space, a high number of time steps and

nonlinear behaviors introduced by ferromagnetic materials. In this context, it is difficult to use a FE model for a co-simulation of an electromagnetic device for a given operating cycle taking into account its electrical or/and mechanical environment. It cannot be applied in real time, as for example, in the "Model in the Loop" context where an experimental bench is composed of real devices and numerical models that exchange data in real-time. Model Order Reduction (MOR) methods can be investigated in order to reduce the computational time required to estimate the solution of the numerical model. In the literature, one of the most popular approaches is the Proper Orthogonal Decomposition (POD) combined with the (Discrete) Empirical Interpolation Method, this *a-posteriori* approach can reduce the size of the time-dependent FE model and the computational time [1, 2]. Based on existing solutions of the FE model for different values of parameters (called snapshots), the POD enables an approximation of the solution of the FE model in a reduced basis [3]. The initial FE system is projected onto the reduced basis determined from the snapshots, decreasing the order of the numerical model to be solved for new parameter values. Nevertheless, in the context of real-time applications, the computational time cannot be sufficiently small. Another method consists in constructing a surrogate model which interpolates directly the FE solution for new parameter values. Different approaches can be used to construct the surrogate model based on the POD. The solution expressed into a reduced basis can be interpolated by an optimization process [4], by polynomial functions [5] or by a Radial Basis Function (RBF) interpolation method [6, 7, 8]. The Proper Generalized Decomposition (PGD) is an *a-priori* approach of MOR which has been also investigated to determine a surrogate model of the FE solution. The PGD method gives an approximation of the FE solution under a separated representation of functions depending on each parameter. After calculation of all functions, the PGD approximation is very fast to be evaluated [9, 10]. In the domain of low frequency electromagnetic modelling, the PGD approach has been used to study a parametric piezoelectric energy harvester [11] or a magneto-quasistatic field problems coupled with external electric circuits [12], for example. The PGD formulation to solve nonlinear problems can be difficult to implement when the number of parameters (source terms, geometric parameters, ...) is significant and also intrusive because it requires modifications of existing FE softwares. To tackle this issue, an approach is to solve the FE model for different values of parameters and to seek for an approximation of the solutions under a separated form, similar to the one given by the intrusive PGD. The PGD method consists in approximating the FE solution to a sum of separable functions (i.e. product of monovariate functions) each depending on one parameter, called modes. Each mode is calculated by an iterative procedure and depends on the previous modes. At each iteration, the functions associated with a mode are defined by a fixed point technique, which requires to solve iteratively different equation systems. From this so-called non-intrusive PGD, an approximate FE solution can be obtained for a wide range of parameters without requiring any additional implementation in the FE model. In the literature, this kind of non-intrusive PGD has been developed for mechanical problem and combined with a nonlinear regression

approach [13, 14, 15, 16].
An alternative approach to the non-intrusive PGD approach is proposed. Based on the solutions of the FE model obtained from different parameter values, a canonical polyadic decomposition is defined in the framework of the PGD. At each iteration, the proposed non-intrusive PGD method computes the discrete values of the functions depending on each parameter and on the previous modes that do not require to solve any equation system. This method is called non-intrusive discrete PGD (d-PGD) approach. The proposed approach is close to the CANDECOMP/PARAFAC (CP) decomposition [18, 19], which is a popular approach to obtain a tensor decomposition of a multidimensional data. The CP decomposition consists in calculating the discrete values of the functions for all modes at the same time. To determine the CP decomposition, an iterative technique based on the Alternating Least Square (ALS) approach is used. The discrete values of functions obtained from the d-PGD or from the CP-ALS undergo an interpolation method in order to define continuous functions depending on each parameter. Then, we obtain a surrogate model of the FE solution which can be evaluated for any parameter value.
In this paper, the d-PGD approach combined with an interpolation method is applied to build a separated representation of the FE solution of a nonlinear magnetostatic problem. Firstly, the numerical model of a nonlinear magnetostatic problem is presented. Secondly, the proposed d-PGD approach combined with a cubic spline interpolation is developed. Finally, a single phase standard transformer and a three-phase inductance are studied. The results obtained from the proposed approach are compared with those given by the CP-ALS and POD-RBF approaches.

2. Nonlinear magnetostatic problem

Let us consider a domain D of boundary Γ ($\Gamma = \Gamma_B \bigcup \Gamma_H$ and $\Gamma_B \bigcap \Gamma_H = 0$) with N_{st} stranded inductors, each supplied by a current i_j (figure (1)).

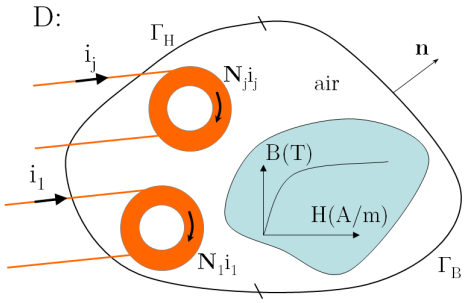


Figure 1: Nonlinear magnetostatic problem.

In magnetostatics, the problem can be described with the following equations

$$\begin{aligned} \mathbf{curl}\ \mathbf{H} &= \sum_{j=1}^{N_{st}} \mathbf{N}_j i_j & (1)\\ div\ \mathbf{B} &= 0 & (2)\\ \mathbf{H} &= \nu(\mathbf{B})\mathbf{B} & (3) \end{aligned}$$

with $\mathbf{B}$ the magnetic flux density, $\mathbf{H}$ the magnetic field, $\mathbf{N}_j$ and i_j the unit current density and the current flowing through the j^{th} stranded inductor and $\nu(\mathbf{B})$ the magnetic reluctivity. For the ferromagnetic material, $\nu(\mathbf{B})$ depends on $\mathbf{B}$ in the nonlinear case. To impose the uniqueness of the solution, boundary conditions must be added

$$\mathbf{B} \cdot \mathbf{n} = 0 \text{ on } \Gamma_B \text{ and } \mathbf{H} \times \mathbf{n} = 0 \text{ on } \Gamma_H \tag{4}$$

with $\mathbf{n}$ the outward unit normal vector. The previous problem can be solved by introducing the vector potential $\mathbf{A}$. From (2), this potential is defined such that $\mathbf{B} = \mathbf{curlA}$ with $\mathbf{A} \times \mathbf{n} = 0$ on Γ_B. Then, based on (1) and (3), the strong formulation to be solved is

$$\mathbf{curl}(\nu(\mathbf{B})\mathbf{curl}\ \mathbf{A}) = \sum_{j=1}^{N_{st}} \mathbf{N}_j i_j. \tag{5}$$

In the $3D$ case, $\mathbf{A}$ and $\mathbf{N}_j$ are discretized using edge and facet elements [17]. The Degrees of Freedom (DoF) of $\mathbf{A}$ are the circulations on the edges of the mesh. Applying the FE method leads to the following system of equations,

$$\mathbf{M}(\mathbf{X})\mathbf{X} = \sum_{j=1}^{N_{st}} \mathbf{F}_j i_j \tag{6}$$

with $i_j \in I_j$ the j^{th} current interval, $\mathbf{M}(\mathbf{X})$ the curl-curl matrix, $\mathbf{F}_j$ the source vector associated with the j^{th} inductor, $\mathbf{X} \in \mathbf{R}^{N_x}$ the vector of DoF of $\mathbf{A}$ and N_x the number of DoF. The previous nonlinear equation system is solved by Newton's method.

The magnetic linkage flux ϕ_j of each inductor j can be expressed as a function of $\mathbf{A}$ by

$$\phi_j = \int_D \mathbf{A} \cdot \mathbf{N}_j dD = \mathbf{F}_j^t \mathbf{X}. \tag{7}$$

3. Non-intrusive discrete Proper Generalized Decomposition

We seek a PGD approximation $\mathbf{X}^{pgd}$ of $\mathbf{X}$ in a separated form. For the sake of simplicity, two inductors flowing by two currents i_1 and i_2 are considered to describe the method, although it can easily extend to a higher number of parameters. $\mathbf{X}^{pgd}$ is expressed by a sum of the products of two scalar functions

$S_{1j}(i_1)$ and $S_{2j}(i_2)$ depending on the currents i_1 and i_2 respectively and of a vector $\mathbf{X}_{Rj}$ of the size of N_x (number of DoF's of $\mathbf{A}$),

$$\mathbf{X}^{pgd}(i_1, i_2) = \sum_{j=1}^{M} \mathbf{X}_{Rj} S_{1j}(i_1) S_{2j}(i_2). \tag{8}$$

Where the term $\mathbf{X}_{Rj} S_{1j}(i_1) S_{2j}(i_2)$ is is the so-called mode and M the number of modes. We are looking for a solution $\mathbf{X}^{pgd}(i_1, i_2)$ that approximate $\mathbf{X}(i_1, i_2)$. $\mathbf{X}^{pgd}$ should be equal or at least be close to $\mathbf{X}(i_1, i_2)$ for a set of values of (i_1, i_2). Then, a set of P vectors $\mathbf{X}(i_1, i_2)$ obtained from the solving of (6) for different values of currents i_1 and i_2, is considered. The number of discrete values for i_1 and i_2 is N_{i_1} and N_{i_2}, respectively and $P = N_{i_1} \times N_{i_2}$. We denote $\mathbf{X}(i_{1k}, i_{2l}) = \mathbf{X}_{kl}$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$. From the P vectors $\mathbf{X}_{kl}$, we seek for the discrete values $X_{1jk} = S_{1j}(i_{1k})$ and $X_{2jl} = S_{2j}(i_{2l})$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$. Hence, the discrete PGD (d-PGD) approximation can be written by

$$\mathbf{X}^{d-pgd}(i_{1k}, i_{2l}) = \mathbf{X}_{kl}^{d-pgd} = \sum_{j=1}^{M} \mathbf{X}_{Rj} X_{1jk} X_{2jl}. \tag{9}$$

The d-PGD approximation seeks to satisfy

$$\mathbf{X}_{kl} - \mathbf{X}_{kl}^{d-pgd} = \mathbf{0} \text{ for } k = 1, ..., N_{i_1} \text{ and } l = 1, ..., N_{i_2}. \tag{10}$$

We define two vectors $[X_{1j1}, ..., X_{1jN_{i_1}}]^t = \mathbf{X}_{1j} \in \mathbf{R}^{N_{i_1}}$ and $[X_{2j1}, ..., X_{2jN_{i_2}}]^t = \mathbf{X}_{2j} \in \mathbf{R}^{N_{i_2}}$. To compute the vectors $\mathbf{X}_{Rj}$, $\mathbf{X}_{1j}$ and $\mathbf{X}_{2j}$ for $j = 1, ..., M$, an iterative enrichment procedure is used. At the n^{th} iteration, the d-PGD approximation is

$$\mathbf{X}_{n,kl}^{d-pgd} = \mathbf{X}_{n-1,kl}^{d-pgd} + \mathbf{X}_{Rn} X_{1nk} X_{2nl} \tag{11}$$

with $\mathbf{X}_{n-1,kl}^{d-pgd}$ the previous approximation with $n-1$ modes. Then, the residue $\mathbf{R}_{n,kl}$ of the n^{th} approximation can be expressed for each (i_{1k}, i_{2l}) by

$$\mathbf{R}_{n,kl} = \mathbf{X}_{kl} - \mathbf{X}_{n,kl}^{d-pgd}. \tag{12}$$

Applying the fixed point technique leads to solving iteratively a problem composed of three sub-problems giving each a solution for $\mathbf{X}_{Rn}$, $\mathbf{X}_{1n}$ and $\mathbf{X}_{2n}$ respectively. The process is repeated until the convergence is reached. In the following, to alleviate the notations, we shall not write any index related to the iteration of the fixed point loop. For the first sub-problem, we assume that the vectors $\mathbf{X}_{1n}$ and $\mathbf{X}_{2n}$ are known. We compute $\mathbf{X}_{Rn}$ in order to have

$$\sum_{k=1}^{N_{i_1}} \sum_{l=1}^{N_{i_2}} X_{1nk} X_{2nl} \mathbf{R}_{n,kl} = \mathbf{0}. \tag{13}$$

Expression (13) can be obtained by minimizing the sum of the square norm of the residue $\mathbf{R}_{n,kl}$ when the entries X_{1nk} and X_{2nl} are fixed (They are supposed

to be known). After developing, we obtain

$$\sum_{k=1}^{N_{i_1}}\sum_{l=1}^{N_{i_2}} X_{1nk}X_{2nl}\mathbf{X}_{Rn}X_{1nk}X_{2nl} =$$
$$\sum_{k=1}^{N_{i_1}}\sum_{l=1}^{N_{i_2}}(X_{1nk}X_{2nl}\mathbf{X}_{kl} - X_{1nk}X_{2nl}\sum_{j=1}^{n-1}\mathbf{X}_{Rj}X_{1jk}X_{2jl}). \tag{14}$$

Finally, the previous equation can be rewritten under a matrix form such as

$$\alpha_{Rn}\mathbf{X}_{Rn} = \mathbf{F}_{Rn} - \sum_{j=1}^{n-1}\alpha_{Rj}\mathbf{X}_{Rj} \tag{15}$$
$$\text{with } \alpha_{Rm} = \mathbf{X}_{1m}^t\mathbf{X}_{1n}\mathbf{X}_{2m}^t\mathbf{X}_{2n} \text{ and } \mathbf{F}_{Rn} = \sum_{k=1}^{N_{i_1}}\sum_{l=1}^{N_{i_2}} X_{1nk}X_{2nl}\mathbf{X}_{kl}.$$

We can observe that since α_{Rn} is a scalar, the vector $\mathbf{X}_{Rn}$ is obtained by some sums and products of vectors, which is quite unexpensive.

For the second sub-problem, we assume that the vectors $\mathbf{X}_{Rn}$ and $\mathbf{X}_{2n}$ are known. We seek for $\mathbf{X}_{1n}$ to impose

$$\sum_{l=1}^{N_{i_2}}\mathbf{X}_{Rn}^t X_{2nl}\mathbf{R}_{n,kl} = \mathbf{0}. \tag{16}$$

Expression (16) is obtained by minimizing the sum of the square norm of the residue $\mathbf{R}_{n,kl}$ where the vector $\mathbf{X}_{Rn}$ and the entries X_{2nl} are supposed to be fixed. Then, we obtain

$$\sum_{l=1}^{N_{i_2}}\mathbf{X}_{Rn}^t X_{2nl}\mathbf{X}_{Rn}X_{1nk}X_{2nl} = \sum_{l=1}^{N_{i_2}}\mathbf{X}_{Rn}^t X_{2nl}\mathbf{X}_{kl} - \mathbf{X}_{Rn}^t X_{2nl}\sum_{j=1}^{n-1}\mathbf{X}_{Rj}X_{1jk}X_{2jl}. \tag{17}$$

Finally, the previous equation can be rewritten under a matrix form such as

$$\alpha_{1n}\mathbf{X}_{1n} = \mathbf{F}_{1n} - \sum_{j=1}^{n-1}\alpha_{1j}\mathbf{X}_{1j} \tag{18}$$
$$\text{with } \alpha_{1m} = \mathbf{X}_{Rm}^t\mathbf{X}_{Rn}\mathbf{X}_{2m}^t\mathbf{X}_{2n},\ \mathbf{F}_{1n} = \mathbf{M}_{2\mathbf{X}}^t\mathbf{X}_{Rn}$$
$$\text{and } [\mathbf{M}_{2\mathbf{X}}]_k = \sum_{l=1}^{N_{i2}} X_{2nl}\mathbf{X}_{kl} \text{ for } k = 1, ..., N_{i1}.$$

$[\mathbf{Y}]_k$ corresponds to the column k of the matrix $\mathbf{Y}$. Again we can see that, since α_{1n} is a scalar, the computation of $\mathbf{X}_{1n}$ is really fast.

For the third sub-problem, we assume that the vectors $\mathbf{X}_{Rn}$ and $\mathbf{X}_{1n}$ are known. In the same way, after minimizing, we obtain $\mathbf{X}_{2n}$ by imposing

$$\sum_{k=1}^{N_{i_1}}\mathbf{X}_{Rn}X_{1nl}\mathbf{R}_{n,kl} = \mathbf{0}. \tag{19}$$

The previous equation can be rewitten under a matrix form such as

$$\alpha_{2n}\mathbf{X}_{2n} = \mathbf{F}_{2n} - \sum_{j=1}^{n-1} \alpha_{2j}\mathbf{X}_{2j} \tag{20}$$
$$\text{with } \alpha_{2m} = \mathbf{X}_{Rm}^t \mathbf{X}_{Rn} \mathbf{X}_{1m}^t \mathbf{X}_{1n},\ \mathbf{F}_{2n} = \mathbf{M}_{1\mathbf{X}}^t \mathbf{X}_{Rn}$$
$$\text{and } [\mathbf{M}_{1\mathbf{X}}]_l = \sum_{k=1}^{N_{i1}} X_{1nl}\mathbf{X}_{kl} \text{ for } l = 1, ..., N_{i2}.$$

The equations (15), (18) and (20) are solved sequentially several times iteratively until convergence of $\mathbf{X}_{Rn}$, $\mathbf{X}_{1n}$ and $\mathbf{X}_{2n}$. In order to stop the iterations of the fixed point loop, three different criteria based on $\mathbf{X}_{Rn}$, $\mathbf{X}_{1n}$ and $\mathbf{X}_{2n}$ between two successive iterations are defined. Then, at the iteration p, we define

$$\epsilon_R = \frac{||\mathbf{X}_{Rn}^p - \mathbf{X}_{Rn}^{p-1}||_2}{||\mathbf{X}_{Rn}^{p-1}||_2},\ \epsilon_1 = \frac{||\mathbf{X}_{i1}^p - \mathbf{X}_{i1}^{p-1}||_2}{||\mathbf{X}_{i1}^{p-1}||_2} \text{ and } \epsilon_2 = \frac{||\mathbf{X}_{i2}^p - \mathbf{X}_{i2}^{p-1}||_2}{||\mathbf{X}_{i2}^{p-1}||_2}. \tag{21}$$

where $\mathbf{Y}_a^p$ and $\mathbf{Y}_a^{p-1}$ represent the quantity $\mathbf{Y}_a$ at the iteration p and $p-1$ respectively. The fixed point loop is stopped when ϵ_R, ϵ_1 and ϵ_2 are simultaneously inferior to the criterion ϵ_{fp} defined by the user or when the number of iterations reach the fixed maximum value n_{fp}^{max} of iterations. We can see that the process for calculating one mode is very fast because only several vector products are required and no solution of equation system is necessary. The number of modes M used for the PGD approximation is not known $a-priori$ by the user. Then, a criterion ϵ_X is defined to stop the enrichment process. After each iteration of the enrichment loop, the relative error ϵ_{kl} is calculated for each value (i_{1k}, i_{2l}) by

$$\epsilon_{kl} = \frac{||\mathbf{X}_{kl} - \mathbf{X}_{kl}^{d-pgd}||_2}{||\mathbf{X}_{kl}||_2}. \tag{22}$$

Then, the criterion ϵ_X is based on the average of the errors

$$\epsilon_X = \frac{1}{P}\sum_{k=1}^{N_{i_1}}\sum_{l=1}^{N_{i_2}} \epsilon_{kl}. \tag{23}$$

The average error of the magnetic linkage flux for the j^{th} inductor can be also defined such as

$$\epsilon_{\phi_j} = \frac{1}{P}\sum_{k=1}^{N_{i_1}}\sum_{l=1}^{N_{i_2}} e_{\phi_{j,kl}} \text{ with } e_{\phi_{j,kl}} = \frac{|\phi_{j,kl} - \phi_{j,kl}^{d-pgd}|}{|\phi_{j,kl}|}. \tag{24}$$

The enrichment process is stopped when ϵ_X is inferior to the fixed criterion ϵ_{ep} or when the number of iterations (i.e. modes) reach the fixed maximum value M^{max} of iterations. Algorithm 1 presents the implemented pseudo-code

Algorithm 1 Non-intrusive d-PGD

Inputs: $\mathbf{X}(i_{1k}, i_{2l})$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$, ϵ_{fp}, n_{fp}^{max}, ϵ_{ep} and M^{max}
Outputs: $\mathbf{X}_{Rj}$, $\mathbf{X}_{1j}$, $\mathbf{X}_{2j}$ for $j = 1, ..., M$

- $M = 0, \epsilon_X = 1$
while $\epsilon_X > \epsilon_{ep}$ and $M < M^{max}$ **do**
  - $M = M + 1$
  - $p = 0$
  - initialisation such as $\mathbf{X}_{2M}^p = \mathbf{X}_{2M-1}$ and $\mathbf{X}_{RM}^p = \mathbf{X}_{RM-1}$
  while ($\epsilon_R > \epsilon_{fp}$ or $\epsilon_1 > \epsilon_{fp}$ or $\epsilon_2 > \epsilon_{fp}$) and $p < n_{fp}^{max}$ **do**
    - $p = p + 1$
    - solve (18) to compute $\mathbf{X}_{1M}^p$ depending on $\mathbf{X}_{2M}^{p-1}$ and on $\mathbf{X}_{RM}^{p-1}$
    - normalization of $\mathbf{X}_{1M}^p$
    - solve (20) to compute $\mathbf{X}_{2M}^p$ depending on $\mathbf{X}_{1M}^p$ and on $\mathbf{X}_{RM}^{p-1}$
    - normalization of $\mathbf{X}_{2M}^p$
    - solve (15) to compute $\mathbf{X}_{RM}^p$ depending on $\mathbf{X}_{1M}^p$ and on $\mathbf{X}_{2M}^p$
    - computation of ϵ_R, ϵ_1 and ϵ_2 by (21)
  end while
  - computation of $\mathbf{X}_{M,kl}^{d-pgd}$ by (11)
  - computation of ϵ_X by (23)
end while

to build a d-PGD approximation $\mathbf{X}^{d-pgd}(i_{1k}, i_{2l})$. The inputs are ϵ_{fp}, ϵ_{ep}, n_{fp}^{max}, M^{max} and $\mathbf{X}(i_{1k}, i_{2l})$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$. With this algorithm, the vectors $\mathbf{X}_{1j}$ and $\mathbf{X}_{2j}$ for $j = 1, ..., M$ are normalized, the magnitude of the d-PGD approximation is supported by $\mathbf{X}_{Rj}$.

After convergence of the process, a d-PGD approximation $\mathbf{X}^{d-pgd}(i_{1k}, i_{2l})$ of the FE solutions $\mathbf{X}(i_{1k}, i_{2l})$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$ is defined. The d-PGD approach can only approximate solution for parameter values corresponding to the FE solutions already calculated. In order to obtain an PGD approximation $\mathbf{X}^{pgd}(i_1, i_2)$ by (8) for any value of currents i_1 and i_2, it is necessary to define the scalar functions $S_{1j}(i_1)$ and $S_{2j}(i_2)$ for $j = 1, ..., M$. Based on the discrete values stored in the vectors $\mathbf{X}_{1j}$ and $\mathbf{X}_{2j}$ of these functions, an interpolation method is used to build the functions $S_{1j}(i_1)$ and $S_{2j}(i_2)$. In order to minimize the Runge phenomenon, an approach based on a cubic spline interpolation is used. Finally, a surrogate model $\mathbf{X}^{pgd}(i_1, i_2)$ of the FE model is obtained which can be evaluated quickly for any value of the currents i_1 and i_2.

From this surrogate model, we can compute approximations of the quantities of interest. Then, the expression of the magnetic flux density depending on the position $\mathbf{x}$ in the studied domain and on the currents is given by

$$\mathbf{B}(\mathbf{x}, i_1, i_2) = \mathbf{curl}\ \mathbf{A}(\mathbf{x}, i_1, i_2) = [\mathbf{curl}\ \mathbf{W}_e(\mathbf{x})]^t \mathbf{X}(i_1, i_2) \approx$$
$$[\mathbf{curl}\ \mathbf{W}_e(\mathbf{x})]^t \mathbf{X}^{pgd}(i_1, i_2) = \sum_{j=1}^{M} \mathbf{B}_j(\mathbf{x}) S_{1j}(i_1) S_{2j}(i_2) \tag{25}$$

with $\mathbf{B}_j(\mathbf{x}) = [\mathbf{curl}\ \mathbf{W}_e(\mathbf{x})]^t \mathbf{X}_{Rj}$ the j^{th} spatial mode of $\mathbf{B}(\mathbf{x}, i_1, i_2)$ and $\mathbf{W}_e(\mathbf{x})$ the vector of edge interpolation functions depending on $\mathbf{x} \in \mathbf{R}^3$ for all edges of the mesh.

For the magnetic linkage flux associated with the k^{th} stranded inductor, we obtain

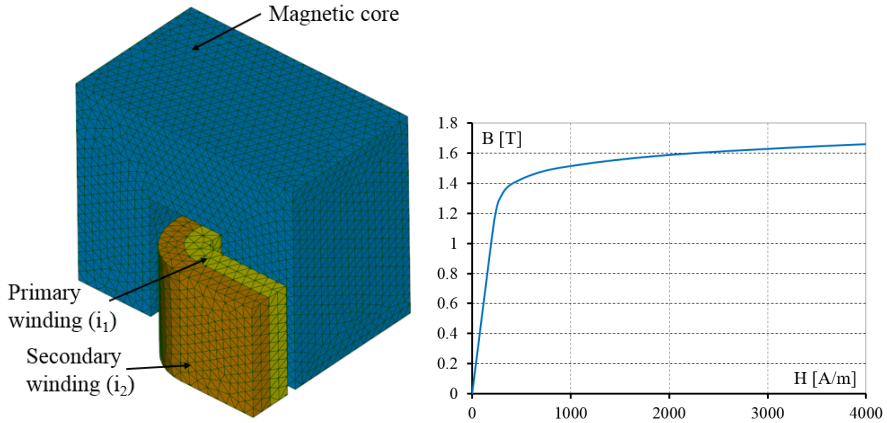
$$\phi_k(i_1, i_2) = \mathbf{F}_k^t \mathbf{X}(i_1, i_2) \approx \mathbf{F}_k^t \mathbf{X}^{pgd}(i_1, i_2) = \sum_{j=1}^{M} \alpha_j S_{1j}(i_1) S_{2j}(i_2) \text{ with } \alpha_j = \mathbf{F}_k^t \mathbf{X}_{Rj}. \tag{26}$$

4. Application

To evaluate the method proposed above, two examples of application are studied. The first one is a single phase transformer and the second one is a three-phase inductance.

4.1. Single phase transformer

Due to the symmetry, only one eighth of a standard single phase transformer is modeled. Figure (2) presents the mesh of the magnetic core and of the windings and the nonlinear magnetic curve respectively. The 3D mesh being composed of 67177 tetrahedron elements, the number of DoF is $N_x = 76663$. The FE model (6) is solved 225 times with $N_{i_1} = N_{i_2} = 15$ equidistributed values for $i_1 \in [0, 1]$ A and for $i_2 \in [-2, 0]$ A.



(a) Mesh (b) Non-linear curve

Figure 2: Single phase EI transformer.

The d-PGD and CP-ALS approaches are applied to find a canonical decomposition of the FE solutions. The CP-ALS method is presented in the appendix A. The parameter values of the d-PGD algorithm are $\epsilon_{fp} = 1e^{-4}$, $n_{fp}^{max} = 40$

and $M^{max} = 150$. For the CP-ALS, the parameter values are $\epsilon_{als} = 1e^{-4}$, $n_{als}^{max} = 2000$, $\beta = 1.3$ and $\delta = 0.3$. Figure (3) presents the errors ϵ_X and ϵ_{ϕ_j} associated with the d-PGD and CP-ALS approaches versus the number of modes M. The convergence of the CP-ALS approach is faster than the one of the d-PGD. Figure (4) gives the computational time of the d-PGD and CP-ALS approaches versus the number of modes M. The d-PGD approach is based on an enrichment process, the calculation of the terms associated with a mode depend on all previous modes. In the case of the CP-ALS, all terms of modes are computed at the same time. Then, the CP-ALS approach is more time consuming than the d-PGD method.

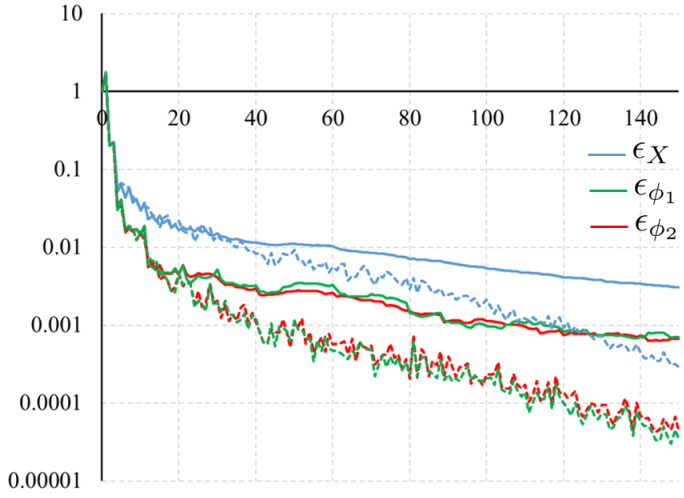


Figure 3: Errors versus the number of modes for the d-PGD (continuous lines) and for the CP-ALS (discontinuous lines).

The canonical decomposition gives a compressed form of data from the solutions $\mathbf{X}$ of the FE model for a value set of (i_{1k}, i_{2l}) for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$. The number of terms extracted from the solutions of the FE model is 76663×225. We define a compression ratio by

$$comp = 1 - \frac{\text{terms for } M \text{ modes of the canonical decomposition}}{\text{terms from the FE model}}. \tag{27}$$

A compression is effective when the ratio *comp* is in the interval $[0, 1]$ and the closer to 1 the higher the compression. The compression ratio gives the percentage of the reduction to the terms number by using a canonical decomposition. The application example with the same assumptions has been studied by an approach based on the POD combined with a RBF interpolation in order to define a surrogate model [8]. The POD-RBF approach is briefly presented in appendix B. Table (1) presents a comparison of different approaches for the same order of the error ϵ_X.

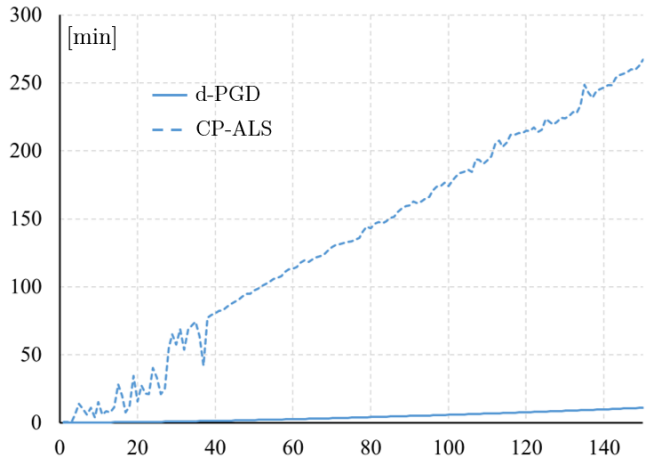


Figure 4: Computational time versus the number of modes for the d-PGD (continuous lines) and for the CP-ALS (discontinuous lines).

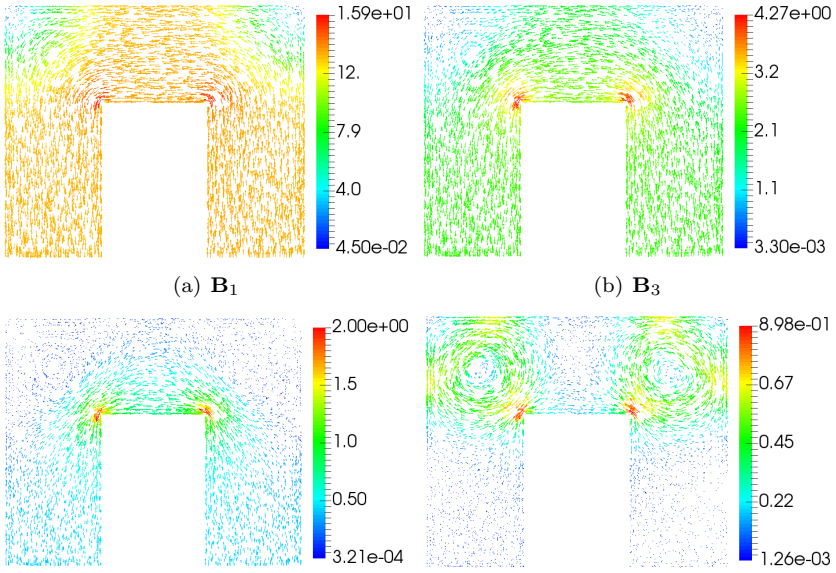
d	ϵ_X	number of modes	computational time [min]	*comp*
d-PGD	$3e^{-3}$	150	12	33%
CP-ALS	$2.9e^{-3}$	80	143	64%
POD-RBF	$8e^{-3}$	N/A	7	80%

Table 1: comparison of different approaches for the single phase transformer

The CP-ALS approach gives the best compression ratio while the POD-RBF approach gives the lower computational time and the largest compression ratio. Nevertheless, the best compromise between the computational time and the error ϵ_X is obtained by the d-PGD approach. If the quantity of interest to be approximated is the magnetic linkage fluxes ϕ_1 and ϕ_2, a good approximation is obtained with a low number of modes. For local quantities such as the magnetic flux density, the number of modes to be considered must be more significant. Since the convergence of the d-PGD approach is quite slow (figure (3)), a good approximation can still be obtained even with a low number of modes. We can see that for the same number of modes ($M = 80$), the error remains acceptable with the d-PGD ($\epsilon_X = 7e - 3$). The computational time required to determine the surrogate model becomes really competitive with the d-PGD being equal to 4.3 min (143 min with the CP-ALS). For $M = 80$ and $\epsilon_X = 7e^{-3}$ with the d-PGD approach, the compression ratio is 64%. Then, the d-PGD approximation is obtained with a reasonable computational time due in particular to the fact that there is no need to solve any equation system.

From the d-PGD approximation (9) and the cubic spline interpolation applied to the vectors $\mathbf{X}_{1j}$ and $\mathbf{X}_{2j}$ for $j = 1, ..., M$ in order to define the functions $S_{1j}(i_1)$ and $S_{2j}(i_2)$, a surrogate model given by the PGD approximation (8) of the FE model is defined for any value of currents i_1 and i_2. Then, we can obtain

an approximation of the magnetic flux density by (25) and of magnetic linkage fluxes by (26). Figure (5) presents the spatial modes $\mathbf{B}_j$ for $j = 1, 3, 5, 7$ of the magnetic flux density.



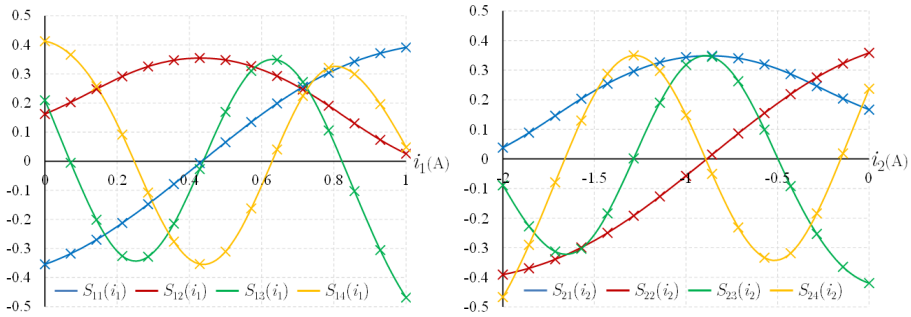
(a) $\mathbf{B}_1$ (b) $\mathbf{B}_3$

(c) $\mathbf{B}_5$ (d) $\mathbf{B}_7$

Figure 5: Distributions of spatial mode $\mathbf{B}_j$

For the first spatial modes $j < 5$, the distributions of $\mathbf{B}_j$ are close to the ones that can be obtained from the FE model with a supply current leading to a physical meaning of such distributions. For the modes $j \geqslant 5$, the distributions of $\mathbf{B}_j$ are mostly located in the corners where the magnetic saturation occurs and which depends highly on the values of the currents. The evolutions of the functions $S_{1j}(i_1)$ and $S_{2j}(i_2)$ for $j = 1, 2, 3, 4$ versus the currents i_1 and i_2 are presented in figure (6). The crosses correspond to the values of $S_{1j}(i_{1k})$ and $S_{2j}(i_{2l})$ with $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$ computed by the d-PGD approximation (9). In the two curve sets of $S_{1j}(i_1)$ and of $S_{2j}(i_2)$, the curves have the same order of magnitude and evolution. This was expected because, for this kind of transformer, the primary and secondary windings create independently a similar magnetic field distribution.

The distribution of the magnetic flux density is approximated by (25), we compute the errors on the approximation of the magnetic linkage flux density distribution and on the magnetic fluxes associated with the windings. We consider two cases, first for the current values (i_1, i_2) used to build the PGD approximation and second for other values of currents. Figure (7) presents the magnetic flux density obtained by (25) for current values ($i_1 = 1$ A, $i_2 = 0$ A) used to build the PGD approximation when the magnetic core is saturated. Figures (8a) and (8b) gives the difference of the magnetic flux density obtained



(a) Functions $S_{1j}(i_1)$ (b) Functions $S_{2j}(i_2)$

Figure 6: Functions $S_{1j}(i_1)$ and $S_{2j}(i_2)$ for $j = 1, 2, 3, 4$ (the crosses correspond to the values of $S_{1j}(i_{1k})$ and $S_{2j}(i_{2l})$ with $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$

.

from the FE model and from the PGD approximation for $M = 80$ and 150. In both cases, the magnitudes of the differences are very small compared with those of the magnetic flux density. When the number of modes increases, the error located close to the internal corners of the magnetic core decreases.

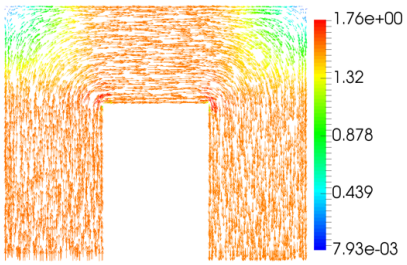
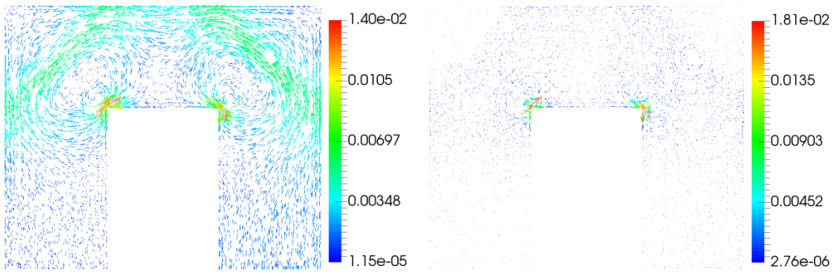


Figure 7: Magnetic flux density **B** [T] when the magnetic core is saturated ($i_1 = 1$ A, $i_2 = 0$ A).

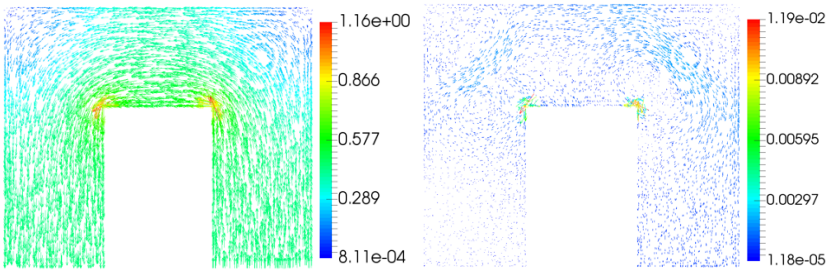
Figures (9a) and (9b) presents the magnetic flux density obtained from the PGD approximation and the difference with $M = 150$ when the magnetic core is not saturated for current values ($i_1 = 0.36$ A, $i_2 = -0.86$ A) used as solution for the PGD approximation. As for the previous case ($i_1 = 1$ A, $i_2 = 0$ A), the errors are small and localized in the corners.

Now, we consider two couples of currents (i_1, i_2) not used to determine the surrogate PGD model (8). The current values are ($i_1 = 0.4$ A, $i_2 = -0.9$ A) and ($i_1 = 0.95$ A, $i_2 = 0$ A). Figures (10) and (11) present the distributions of the magnetic flux density obtained by (25) and the differences for both couples of current. As for the previous studies, the maximum of the errors are located in the corners and the magnitudes of the differences are small. In term of global quantities, the magnetic linkage fluxes can be approximated by (26). Then, for ($i_1 = 0.4$ A, $i_2 = -0.9$ A) and ($i_1 = 0.4$ A,$i_2 = -0.9$ A), we obtain ($\phi_1 = -5.57$



(a) $M = 80$ (b) $M = 150$

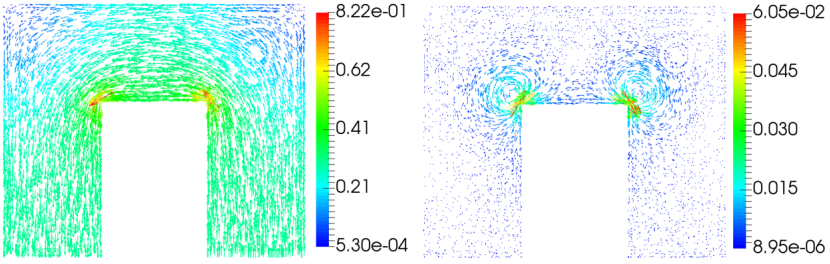
Figure 8: Difference of the magnetic flux density [T] when the magnetic core is saturated ($i_1 = 1$ A, $i_2 = 0$ A).



(a) Magnetic flux density **B** [T] (b) Difference of the magnetic flux density [T]

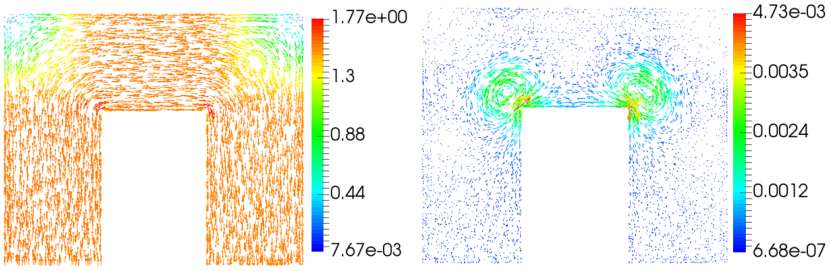
Figure 9: The magnetic core is not saturated ($i_1 = 0.36$ A, $i_2 = -0.86$ A).

Wb, $\phi_2 = -2.73$ Wb) and ($\phi_1 = 0.29$ Wb, $\phi_2 = 0.14$ Wb) respectively. The maximal error on the quantity is about 0.32%.



(a) Magnetic flux density **B** [T] (b) Difference of the magnetic flux density [T]

Figure 10: Distribution of the magnetic flux density [T] and of the error for ($i_1 = 0.4$ A, $i_2 = -0.9$ A).



(a) Magnetic flux density $\mathbf{B}$ [T]

(b) Difference of the magnetic flux density [T]

Figure 11: Distribution of the magnetic flux density and of the error for ($i_1 = 0.95$ A, $i_2 = 0$ A).

4.2. Three-phase inductance

Due to the symmetry, one quarter of the three-phase inductance is modeled. Figure (12) presents the mesh of the magnetic core and of the windings. We consider the nonlinear magnetic curve presented on figure (2-b). The 3D mesh is composed of 66382 tetrahedron elements, the number of DoF is $N_x = 75584$. The FE model (6) is solved 2197 times with 13 equidistributed values N_{i_1}, N_{i_2} and N_{i_3} with the same current interval $I = [-6, 6]$ A for all currents. The number of terms extracted from the solutions of the FE model is 75584×2197 and the computational time is about 57.5 h.

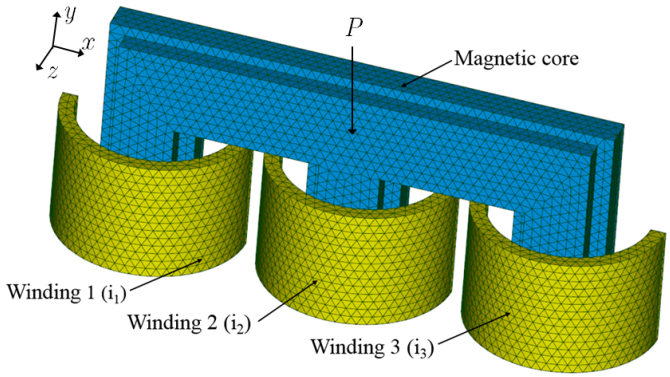


Figure 12: Mesh of the three-phase inductance.

Figure (13) presents the errors of the d-PGD approximation (9) and of magnetic linkage fluxes versus the number of modes.

The application example with the same assumptions has been studied by the POD-RBF approach in order to define a surrogate model [8]. Table (2) presents a comparison of both approaches for the same order of the error ϵ_X with $350 \times (75584 + 13 + 13 + 13)$ terms computed for the d-PGD approximation.

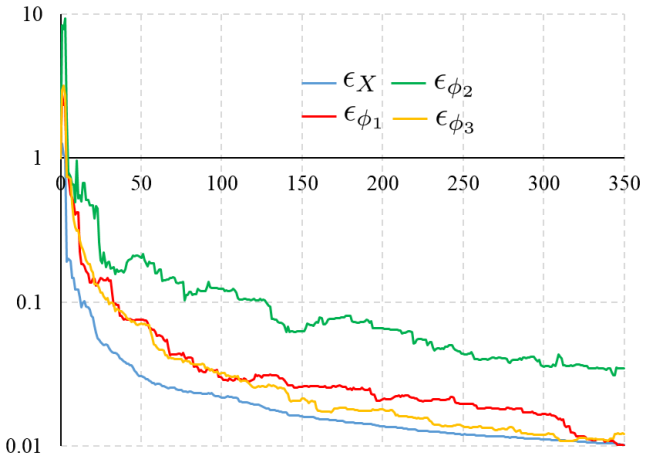


Figure 13: Errors versus the number of modes.

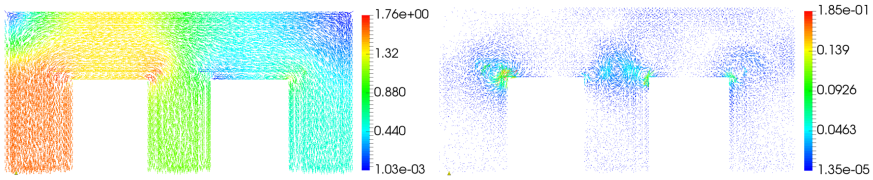
The computational time and the compression ratio are the same order for both approaches. Even if the errors ϵ_X are close, the errors of the magnetic fluxes from the d-PGD approach are lower than those from the POD-RBF.

	d-PGD	POD-RBF
ϵ_X	$1e^{-2}$	$1.41e^{-2}$
ϵ_{ϕ_1}	$1.01e^{-2}$	$4.38e^{-2}$
ϵ_{ϕ_2}	$3.47e^{-2}$	$8.41e^{-2}$
ϵ_{ϕ_3}	$1.21e^{-2}$	$6.38e^{-2}$
number of modes	350	N/A
computational time [h]	5.3	5
comp	84%	94%

Table 2: Comparison of different approaches for the three-phase inductance.

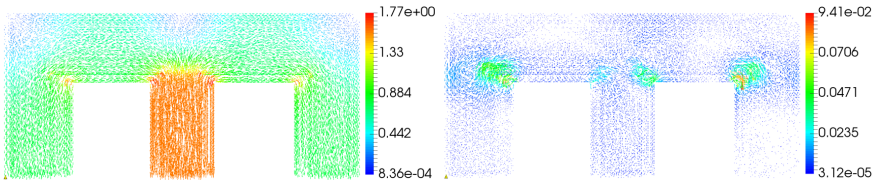
Figures (14) and (15) present the distribution of the magnetic flux density obtained from the d-PGD approximation (9) and the difference when all windings are supplied such as the sum of currents is null ($i_1 = 6$ A, $i_2 = -3$ A, $i_3 = -3$ A) and when only the winding 2 is supplied ($i_1 = 0$ A, $i_2 = 6$ A, $i_3 = 0$ A). These two sets of currents have been used as solutions to build the d-PGD approximation. For both cases, the errors are located at the corners of the geometry with smaller magnitudes than those of the magnetic flux density.

From the d-PGD approximation, a surrogate model (8) of the FE model is built to give an approximation of the FE solution for any value of currents (i_1, i_2, i_3). Then, we can compute an approximation of the magnetic flux density by (25) and of magnetic fluxes by (26). To evaluate the accuracy of the surrogate model, three cases are considered. In the first two cases, a balanced three phase



(a) Magnetic flux density **B** [T] (b) Difference of the magnetic flux density [T]

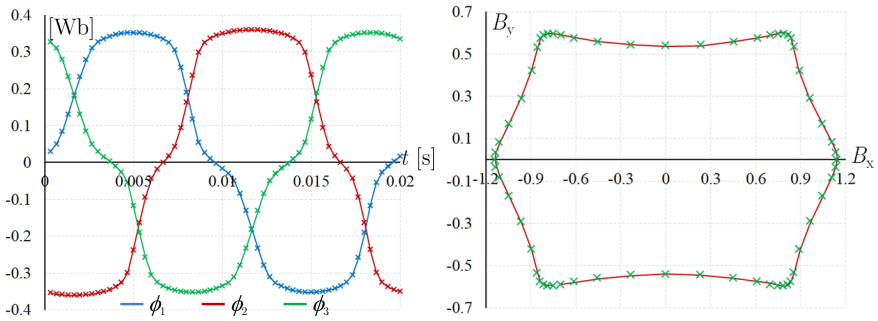
Figure 14: Three-phase inductance when all winding are supplied



(a) Magnetic flux density **B** [T] (b) Difference of the magnetic flux density [T]

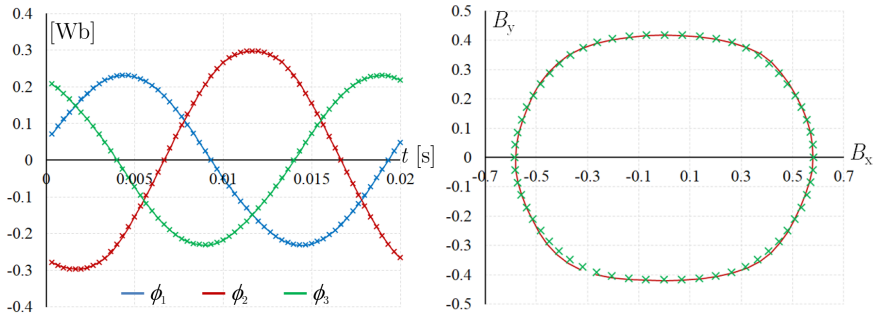
Figure 15: Three-phase inductance when only the winding 2 is supplied.

current supply is applied such as $i_1(t) = I_M sin(\omega t)$, $i_2(t) = I_M sin(\omega t - \frac{2\pi}{3})$ and $i_2(t) = I_M sin(\omega t - \frac{4\pi}{3})$ with ω the angular frequency and $I_M = 6$ A in order to saturate the magnetic core of the three-phase inductance in the first case and $I_M = 2$ A in the second case. In the third case, an example of failure is studied where the first winding is not supplied $i_1 = 0$ A and the two others are supplied such as $i_2(t) = -i_3(t) = I_M sin(\omega t - \frac{2\pi}{3})$ with $I_M = 4$ A. In all three cases, the angular frequency is fixed to $\omega = 2\pi f$ with $f = 50$ Hz and the time simulation corresponds to one period with 60 time steps equidistributed. The quantities of interest are the magnetic linkage fluxes of the windings computed by (26) and the magnetic flux density at the point P located on the figure (12) and calculated by (25). The results obtained from the surrogate model are compared with those given by the FE model. Figures (16), (17) and (18) present the time evolution of the magnetic linkage fluxes of the windings and the magnetic flux density in the plan (x, y) at the point P obtained from the PGD surrogate model. On these figures, the results given by the FE model are presented by the curves with crosses. For all cases, the results calculated with the surrogate model are in good agreement with the values obtained from the FE model. In terms of computational time, the surrogate model and the FE model require about 16 s and 80 min respectively to compute the magnetic linkage fluxes and the magnetic flux density at the point P for 60 time steps.



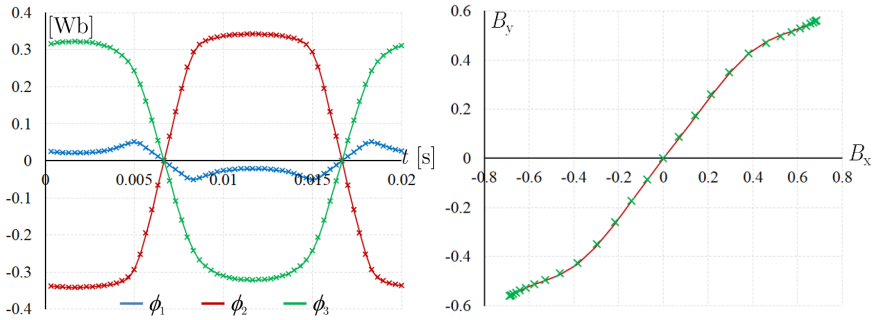
(a) Magnetic linkage fluxes versus the time (b) Magnetic flux density on the point P

Figure 16: Case 1: a balanced three phases current supply is applied such as $i_1(t) = I_M sin(\omega t)$, $i_2(t) = I_M sin(\omega t - \frac{2\pi}{3})$ and $i_2(t) = I_M sin(\omega t - \frac{4\pi}{3})$ with $I_M = 6$ A.



(a) Magnetic linkage fluxes versus the time (b) Magnetic flux density on the point P

Figure 17: Case 2: a balanced three phases current supply is applied such as $i_1(t) = I_M sin(\omega t)$, $i_2(t) = I_M sin(\omega t - \frac{2\pi}{3})$ and $i_2(t) = I_M sin(\omega t - \frac{4\pi}{3})$ with $I_M = 2$ A.



(a) Magnetic linkage fluxes versus the time (b) Magnetic flux density on the point P

Figure 18: Case 3: default situation where the first winding is not supplied $i_1 = 0$ A and the two others are supplied such as $i_2(t) = -i_3(t) = I_M sin(\omega t - \frac{2\pi}{3})$ with $I_M = 4$ A.

5. Conclusion

From solutions of a FE model for different values of parameters, a surrogate model has been built based on a non-intrusive discrete PGD approach combined with an interpolation method. The proposed approach has been developed in the case of a nonlinear magnetostatic problem considering the currents of the windings as parameters. Two 3D application examples have been studied. The results obtained from the proposed non-intrusive PGD approach have been compared with those given from the CP-ALS and POD-RBF approaches. The proposed approach is a good trade-off in terms of accuracy, computational time and compression ratio compared with both approaches. It appears through the two examples of application that the PGD surrogate model is obtained with reasonable computational times compared to those required to solve the FE model for all sets of parameters. Moreover, the PGD surrogate model is very fast to be evaluated in order to obtain an approximate FE solution and it can be used in a real-time application or in co-simulation study to simulate a magnetic device in its environment.

Appendix A. CP-ALS decomposition

The CANDECOMP/PARAFAC or canonical polyadic (CP) decomposition [18] [19] is a popular approach to obtain a tensor decomposition of a multi-dimensional data. One of the most popular methods used to compute a CP decomposition is the alternating least squares (ALS) approach which consists in solving a sequence of linear least squares problems. The non-intrusive PGD approach introduced in the section 3 is a method close to the CP-ALS decomposition approach. In the following, the CP-ALS approach is briefly presented in the same context. The PGD approximation (9) is based on P solutions $\mathbf{X}(i_1, i_2)$ of size N_x such as $\mathbf{X}(i_{1k}, i_{2l}) = \mathbf{X}_{kl}$ for $k = 1, ..., N_{i_1}$ and $l = 1, ..., N_{i_2}$. To apply the CP-ALS method, the set of solutions is reordered as a third-order tensor $\mathcal{X}$ of size $N_x \times N_{i_1} \times N_{i_2}$. The CP decomposition consists in determining an approximation of $\mathcal{X}$ such as

$$\mathcal{X} \approx \sum_{j=1}^{M} \mathbf{X}_{Rj} \circ \mathbf{X}_{1j} \circ \mathbf{X}_{2j} \tag{A.1}$$

with M the rank of the decomposition corresponding to the number of modes, $\circ$ the vector outer product, $\mathbf{X}_{1j} = [X_{1j1}, ..., X_{1jN_{i_1}}]^t$ and $\mathbf{X}_{2j} = [X_{2j1}, ..., X_{2jN_{i_2}}]^t$. The sets of vectors $\mathbf{X}_{Rj}$, $\mathbf{X}_{1j}$ and $\mathbf{X}_{2j}$ for $j = 1, ..., M$ are arranged in order to define three matrices such as $\mathbf{X}_R = [\mathbf{X}_{R1}\, \mathbf{X}_{R2} \ldots \mathbf{X}_{RM}]$, $\mathbf{X}_1 = [\mathbf{X}_{11}\, \mathbf{X}_{12} \ldots \mathbf{X}_{1M}]$ and $\mathbf{X}_2 = [\mathbf{X}_{21}\, \mathbf{X}_{22} \ldots \mathbf{X}_{2M}]$. To apply the CP-ALS approach, the tensor $\mathcal{X}$ is reordered by three different matrices $\mathbf{M}_R$, $\mathbf{M}_1$ and $\mathbf{M}_2$ of size $N_x \times (N_{i_1} N_{i_2})$, $N_{i_1} \times (N_x N_{i_2})$ and $N_{i_2} \times (N_x N_{i_2})$ respectively. Then, $\mathbf{X}_R$, $\mathbf{X}_1$ and $\mathbf{X}_2$ are

computed by repeating the solution of three equations

$$\min_{\mathbf{X}_R} ||\mathbf{M}_R - \mathbf{X}_R(\mathbf{X}_2 \odot \mathbf{X}_1)^t||_F$$
$$\Rightarrow \mathbf{X}_R = \mathbf{M}_R(\mathbf{X}_2 \odot \mathbf{X}_1)(\mathbf{X}_2^t\mathbf{X}_2 * \mathbf{X}_1^t\mathbf{X}_1)^\dagger \quad (A.2)$$
$$\min_{\mathbf{X}_1} ||\mathbf{M}_1 - \mathbf{X}_1(\mathbf{X}_2 \odot \mathbf{X}_R)^t||_F$$
$$\Rightarrow \mathbf{X}_1 = \mathbf{M}_1(\mathbf{X}_R \odot \mathbf{X}_2)(\mathbf{X}_R^t\mathbf{X}_R * \mathbf{X}_2^t\mathbf{X}_2)^\dagger \quad (A.3)$$
$$\min_{\mathbf{X}_2} ||\mathbf{M}_2 - \mathbf{X}_2(\mathbf{X}_1 \odot \mathbf{X}_R)^t||_F$$
$$\Rightarrow \mathbf{X}_2 = \mathbf{M}_2(\mathbf{X}_R \odot \mathbf{X}_1)(\mathbf{X}_R^t\mathbf{X}_R * \mathbf{X}_1^t\mathbf{X}_1)^\dagger \quad (A.4)$$

with $\odot$ the Khatri-Rao product and $\mathbf{Y}^\dagger$ the pseudo inverse of $\mathbf{Y}$. Algorithm 2 presents the implemented pseudo-code of the CP-ALS approach with, as inputs, M the number of modes, ϵ_{als} a criterion used to stop the iterations and n_{als}^{max} the maximal number of iterations. The vectors of $\mathbf{X}_1$ and $\mathbf{X}_2$ are normalized, the magnitudes are supported by the vectors of $\mathbf{X}_R$.

Algorithm 2 CP-ALS approach

Input: $\mathcal{X}$, ϵ_{als} and n_{als}^{max}
Output: $\mathbf{X}_R$, $\mathbf{X}_1$, $\mathbf{X}_2$

- $p = 0, \epsilon_{\mathcal{X}} = 1$
- **while** $\epsilon_{\mathcal{X}} > \epsilon_{als}$ and n_{als}^{max} **do**
  - $p = p + 1$
  - solve (A.3) to compute $\mathbf{X}_1^p$ depending on $\mathbf{X}_2^{p-1}$ and on $\mathbf{X}_R^{p-1}$
  - normalization of the vectors of $\mathbf{X}_1^p$
  - solve (A.4) to compute $\mathbf{X}_2^p$ depending on $\mathbf{X}_1^p$ and on $\mathbf{X}_R^{p-1}$
  - normalization of the vectors of $\mathbf{X}_2^p$
  - solve (A.2) to compute $\mathbf{X}_R^p$ depending on $\mathbf{X}_1^p$ and on $\mathbf{X}_2^p$
  - computation of $\epsilon_{\mathcal{X}}$ such as $\epsilon_{\mathcal{X}} = ||\mathbf{M}_1 - \mathbf{X}_1^p(\mathbf{X}_2^p \odot \mathbf{X}_R^p)^t||_F$
- **end while**

It is well known that the convergence of the CP-ALS can be very slow. Then, different approaches have been proposed in the literature in order to improve the convergence. In [20], [21] and [22], the principle is to compute $\mathbf{X}_R^p$, $\mathbf{X}_1^p$ and $\mathbf{X}_2^p$ at each iteration p based on the approximations $\mathbf{X}_R^{ex}$, $\mathbf{X}_1^{ex}$ and $\mathbf{X}_2^{ex}$. These approximations are calculated by a line search approach depending on $\mathbf{X}_R^{p-l}$, $\mathbf{X}_1^{p-l}$ and $\mathbf{X}_2^{l-p}$ with $l = 1, 2, 3$ such as

$$\mathbf{X}_R^{ex} = \mathbf{X}_R^{p-2} + \beta\mathbf{G}_R^p \quad (A.5)$$
$$\text{with } \mathbf{G}_R^p = (1+\delta)\mathbf{X}_R^{p-1} - (1+2\delta)\delta\mathbf{X}_R^{p-2} + \delta\mathbf{X}_R^{p-3}$$
$$\mathbf{X}_1^{ex} = \mathbf{X}_1^{p-2} + \beta\mathbf{G}_1^p \quad (A.6)$$
$$\text{with } \mathbf{G}_1^p = (1+\delta)\mathbf{X}_1^{p-1} - (1+2\delta)\delta\mathbf{X}_1^{p-2} + \delta\mathbf{X}_1^{p-3}$$
$$\mathbf{X}_2^{ex} = \mathbf{X}_2^{p-2} + \beta\mathbf{G}_2^p \quad (A.7)$$
$$\text{with } \mathbf{G}_2^p = (1+\delta)\mathbf{X}_2^{p-1} - (1+2\delta)\delta\mathbf{X}_2^{p-2} + \delta\mathbf{X}_2^{p-3}$$

with β and δ two parameters. Algorithm 3 presents the implemented pseudo-code of the CP-ALS approach combined with the line search approach.

Algorithm 3 CP-ALS approach combined with a line search approach

Input: $\mathcal{X}$, ϵ_{als} and n_{als}^{max}
Output: $\mathbf{X}_R$, $\mathbf{X}_1$, $\mathbf{X}_2$

- $p = 0, \epsilon_{\mathcal{X}} = 1$
while $\epsilon_{\mathcal{X}} > \epsilon_{als}$ and n_{als}^{max} **do**
  - $p = p + 1$
  - compute $\mathbf{X}_2^{ex}$ by (A.7)
  - normalization of the vectors of $\mathbf{X}_2^{ex}$
  - compute $\mathbf{X}_R^{ex}$ by (A.5)
  - solve (A.3) to compute $\mathbf{X}_1^p$ depending on $\mathbf{X}_2^{ex}$ and on $\mathbf{X}_R^{ex}$
  - normalization of the vectors of $\mathbf{X}_1^p$
  - solve (A.4) to compute $\mathbf{X}_2^p$ depending on $\mathbf{X}_1^p$ and on $\mathbf{X}_R^{ex}$
  - normalization of the vectors of $\mathbf{X}_2^p$
  - solve (A.2) to compute $\mathbf{X}_R^p$ depending on $\mathbf{X}_1^p$ and on $\mathbf{X}_2^p$
  - computation of $\epsilon_{\mathcal{X}}$ such as $\epsilon_{\mathcal{X}} = ||\mathbf{M}_1 - \mathbf{X}_1^p(\mathbf{X}_2^p \odot \mathbf{X}_R^p)^t||_F$
end while

At each iteration p, the optimal value of the parameter β can be defined by computing the value which minimizes $||\mathbf{M}_1 - \mathbf{X}_1^{ex}(\mathbf{X}_2^{ex} \odot \mathbf{X}_R^{ex})^t||_F$.

Appendix B. Surrogate model based on POD combined with RBF interpolation

The determination of a surrogate model based on the POD approach combined with the RBF interpolation method is briefly presented in the context of the section 3. This approach is detailed in [6, 7, 8]. A POD-RBF surrogate model is based on a set of P vectors $\mathbf{X}(i_1, i_2)$ for different values of currents vector (i_1, i_2). Then, an approximation $\mathbf{X}^{pod-rbf}(i_1, i_2)$ is sought under the form

$$\mathbf{X}^{pod-rbf}(i_1, i_2) = \sum_{j=1}^{M} \psi_j g_j(i_1, i_2) \tag{B.1}$$

with $\psi_j \in \mathbf{R}^{N_x}$, $g_j(i_1, i_2)$ a scalar function and M the number of modes. Based on the Singular Value Decomposition (SVD) of the FE solutions matrix $\mathbf{M}_X = [\mathbf{X}(i_{11}, i_{21}), \mathbf{X}(i_{12}, i_{22}), ..., \mathbf{X}(i_{1P}, i_{2P})] = \mathbf{USV^t}$, the reduced basis $\mathbf{\Psi} = [\psi_l, \psi_2, ..., \psi_M]$ of size $N_x \times M$ is defined by the M first columns of $\mathbf{U}$. The truncation is obtained by taking the vectors ψ_j corresponding to the M most significant singular values of $\mathbf{S}$. To determine the functions $g_j(i_1, i_2)$ for $j = 1, ..., M$, the FE solutions matrix expressed into the reduced basis $\mathbf{\Psi}$ by $\mathbf{M}_{X_r} = \mathbf{S}_{(1:M,1:M)}\mathbf{V}^t_{(1:P,1:M)}$ is used. Each line of $\mathbf{M}_{X_r}$ corresponds to the discrete values of $g_j(i_{1l}, i_{2l})$ for $l = 1, ..., P$. From these values, an RBF interpolation is performed in order to express each function $g_j(i_1, i_2)$ for any current

vector (i_1, i_2) under the following form

$$g_j(i_1, i_2) = \sum_{l=1}^{P} \alpha_{jl} \phi_l(i_1, i_2) \tag{B.2}$$

with α_{jl} a coefficient and ϕ_l a radial function depending on the Euclidean distance between (i_1, i_2) and (i_{1l}, i_{2l}) and on a parameter called "shape parameter". For the radial functions, multiquadric functions are used. Based on a FE solutions matrix composed of Q vectors, a Greedy algorithm can be used to select the P most significant FE solutions in order to optimize the reduced basis [8].

References

[1] S. Chaturantabut, D. C. Sorensen, "Nonlinear Model Reduction via Discrete Empirical Interpolation", *SIAM J. Sci. Comput.*, vol. 32, no. 5, pp.2737-2764, 2010.

[2] T. Henneron and S. Clénet, "Model-Order Reduction of Multiple-Input Non-Linear Systems Based on POD and DEI Methods," *IEEE Trans. Magn.*, vol. 51, no. 3, pp. 1-4, March 2015.

[3] J. Lumley, 'The structure of inhomogeneous turbulence', *Atmospheric Turbulence and Wave Propagation*, A.M. Yaglom and V.I. Tatarski., pp. 221-227, 1967.

[4] R.R. Rama, S. Skatulla, C. Sansour, "Real-time modelling of the heart using the proper orthogonal decomposition with interpolation", *VI International Conference on Computational Bioengineering*, ICCB 2015, 2015.

[5] M. Farzam Far, F. Martin, A. Belahcen, L. Montier, T. Henneron, "Orthogonal Interpolation Method for Order Reduction of a Synchronous Machine Model," *IEEE Trans. Mag.*, vol. 54(2), pp. 1-6, 2018.

[6] W.P. Adamczyk , Z. Ostrowski, Z. Buli?ski, A. Ryfa, "Application of POD-RBF technique for retrieving thermal diffusivity of anisotropic material", *9th International Conference on Inverse Problems in Engineering*, 2017.

[7] V. Buljak, "Proper orthogonal decomposition and radial basis functions algorithm for diagnostic procedure based on inverse analysis", *FME Transaction*, vol. 38, 2010.

[8] T. Henneron, A. Pierquin and S. Clénet, "Surrogate Model Based on the POD Combined With the RBF Interpolation of Nonlinear Magnetostatic FE Model", *IEEE Trans. Magn.*, vol. 56, no. 1, pp. 1-4, Jan. 2020.

[9] F. Chinesta, A. Ammar, E. Cueto, "Recent Advances and New Challenges in the Use of the Proper Generalized Decomposition for Solving Multidimensional Models", *Archives of Computational Methods in Engineering*, vol. 17(4), pp. 327-350, 2010.

[10] F. Chinesta , R. Keunings, A. Leygue, "The proper generalized decomposition for advanced numerical simulations", A primer: Springerbriefs. New York: Springer, 2014.

[11] Z. Qin, H. Talleb, S. Yan, X. Xu,Z. Ren, "Application of PGD on Parametric Modeling of a Piezoelectric Energy Harvester," *IEEE Trans. Magn.*, vol. 52, no. 11, pp. 1-11, Nov. 2016, 7210211.

[12] T. Henneron, S. Clénet, "Proper Generalized Decomposition Method Applied to Solve 3-D Magnetoquasi-Static Field Problems Coupling With External Electric Circuits," *IEEE Trans. Magn.*, vol. 51, no. 6, pp. 1-10, June 2015.

[13] R. Ibáñez, E. Abisset-Chavanne, A. Ammar, D. González, A. Huerta, J-L Duval, F. Chinesta, "A Multidimensional Data-Driven Sparse Identification Technique: The Sparse Proper Generalized Decomposition", *Complexity*, vol. 2018, Article ID 5608286, 11 pages, 2018.

[14] Y. Lu, N. Blal, A. Gravouil, "Multi-parametric space-time computational vademecum for parametric studies: Application to real time welding simulations", *Finite Elements in Analysis and Design*, Vol. 139, pp. 62-72, 2018.

[15] A. Sancarlos, V. Champaney, J. Duval, E. Cueto, F. Chinesta, "PGD-based advanced nonlinear multiparametric regressions for constructing metamodels at the scarce-data limit",*Computer Science*, ArXiv, 2021.

[16] X. Zou, M. Conti, P. Dez, F. Auricchio, "A nonintrusive proper generalized decomposition scheme with application in biomechanics", *Int J Numer Meth Engng.*, vol 113, pages 230 251, 2018.

[17] A. Bossavit, "Whitney forms: A class of finite elements for three-dimensional computations in electromagnetism", *Science, Measurement and Technology, IEE Proceedings A* 135(8):493 - 500, December 1988.

[18] J. D. Carroll, J. J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition", Psychometrika, 35 (1970), pp. 283319.

[19] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an explanatory multi-modal factor analysis", UCLA Working Papers in Phonetics, 16 (1970).

[20] M. Rajih, P. Comon, R. Harshman, "Enhanced Line Search : A Novel Method to Accelerate PARAFAC", SIAM Journal on Matrix Analysis and Applications, Society for Industrial and Applied Mathematics, 2008, 30 (3), pp.1148-1171.

[21] D. Nion, L. De Lathauwer, "An enhanced line search scheme for complex-valued tensor decompositions. Application in DS-CDMA", Signal Processing, Volume 88, Issue 3, 2008, pp. 749-755.

[22] Y. Chen, D. Han and L. Qi, "New ALS Methods With Extrapolating Search Directions and Optimal Step Size for Complex-Valued Tensor Decompositions," in IEEE Transactions on Signal Processing, vol. 59, no. 12, pp. 5888-5898, Dec. 2011.