



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/25264>

To cite this version :

T. DELAGNES, Thomas HENNERON, Stephane CLENET, M. FRATILA, J.P. DUCREUX - Comparison of reduced basis construction methods for Model Order Reduction, with application to non-linear low frequency electromagnetics - Mathematics and Computers in Simulation - Vol. 211, p.470-488 - 2023

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



Comparison of reduced basis construction methods for Model Order Reduction, with application to non-linear low frequency electromagnetics

T. Delagnes ^{1,2}, T. Henneron ¹, S. Clenet ¹, M. Fratila ² and J.P. Ducreux ²

¹ Univ. Lille, Centrale Lille, Arts et Metiers ParisTech, HEI, EA 2697 - L2EP, F-59000 Lille, France

E-mail: theo.delagnes@edf.fr

² EDF R&D,ERMES, 7 Boulevard Gaspard Monge, 91120 Palaiseau, France

Abstract

Numerical simulation is more and more used during the design stage of a manufactured product in order to optimize its performances. However, it is often too time consuming, particularly when it's used to solve optimization problems, preventing an intensive usage. A-posteriori Model Order Reduction methods can be very effective to shorten the computational time. An approximated solution is then sought in a space of small dimension defined by a reduced basis. The accuracy of such methods is highly dependent on the choice of the reduced basis, extracted from preliminary numerical simulation. The method usually applied to construct such reduced basis is based on the Singular Value Decomposition (SVD), which can be time consuming, and is not adapted when a large collection of preliminary numerical simulations must be used to construct the basis. An alternative to this approach has been proposed recently with the Maximum Entropy Snapshot Sampling (MESS) method. In this paper, we propose to compare these methods with other approaches usually used for clustering or data classification based on vectors distance calculation, like the Centroidal Voronoi Tessellation (CVT), Density Based Spatial Clustering of Applications with Noise (DBSCAN), and Clustering Using Representatives (CURE). The methods are compared on a complex and realistic nonlinear problem in low frequency electromagnetics. The quality of the reduced bases obtained by the different methods are compared. Then, field distributions and global quantities, like eddy current losses and magnetic energy, are computed from the reconstructed results, to further analyze the quality of the reduced bases.

Numerical simulation, Finite Element method, Model order reduction, Reduced basis, Electromagnetics

Introduction

Numerical simulation is a widely used tool in industry as well as in research, as it allows having access to any quantity of interest, without the constraint of conducting real life experiments. However, simulations can be extremely time consuming, especially in the case of parametric studies or optimization problems, when it is necessary to run many simulations with varying parameters. Hence, a reduction of the computation time of the simulation is required, accepting a loss of the precision on the

result. In this context, there has been an intensive development of Model Order Reduction techniques in the past decades.

Model Order Reduction field gathers a collection of methods to reduce the computation time of numerical simulations. Two kinds of reduction methods exist: *a priori* and *a posteriori*. *A priori* methods are automated algorithms, which do not require any specific knowledge on the problem to solve. This is a considerable asset in terms of ease of use. However, these methods are not necessarily easy to implement and can have convergence issues when applied to non-linear problems. The Proper Generalized Decomposition (PGD) [1][2] and the Reduced Basis method (RB) [3] are examples of such methods. On the other hand, *a posteriori* methods require preliminary simulation results of the problem to solve, in order to derive the reduced model. From these preliminary simulation results, called snapshots, a reduced basis can be constructed, on which the problem solution and the matrix system are projected. The number of unknowns of the problem can be drastically reduced leading to an equation system of small size. With this reduction process, an accuracy loss appears on the solution, which is highly related to the choice of the reduced basis. In fact, to obtain a good approximation with the reduced model, its solution must be close to the solution of the initial problem projected in the space defined by the reduced basis. The challenges when using *a posteriori* reduction methods are the choice of the preliminary simulation results, as well as the method used to construct the reduced basis. Concerning the choice of preliminary results, it greatly depends on the targeted operating range of validity of the reduced model, and thus often necessitates a significant technical background to be correctly defined. In the case of low frequency electromagnetic problems, a methodology was proposed based on a well-known standard approach using typical tests [4][5], which showed great accuracy of the reduced model as well as the need of an important quantity of preliminary results. Thereby, once the right preliminary results are generated, one can start to construct the reduced basis by applying numerical techniques. The paper focuses on the comparison of these different techniques.

To construct a reduced basis using *a posteriori* reduction methods, one needs to find the redundancy information in the snapshots, with clustering techniques [6] or Singular Value Decomposition (SVD) [7]. SVD is the most used method to construct a reduced basis from a set of snapshots, as it allows extracting independent vectors containing the information of the snapshots, ranked in terms of energy. Clustering methods, on the other hand, allow selecting the snapshots for the construction of a reduced basis using various techniques. Centroidal Voronoi Tessellation (CVT) [8], for example, gathers in small sets snapshots that are close to each other in terms of the Frobenius norm, and extract the basis by computing the centers of each set. We can find applications of such methods for reduced model construction in robotics [9], design optimization [10], and power network analysis [11]. In computational physics, researchers use this kind of methods, in fluid dynamics [12][13], mechanical systems [14][15] and electromagnetics [16][17][18]. Recently, model order reduction schemes have been coupled with uncertainty quantification methods to find an approximation of the solution of ordinary differential equations [19], and to nonlinear regression techniques and Optimal Transport to find parametric solutions of fluid dynamics problem [20]. Also, reduction methods have been used to simulate compressible flow with particular boundary conditions [21], and to represent in 3D the behavior of multilayered components in microwave fields [22]. On the other hand, clustering methods are used in Neural Network applied to electrical load forecasting for demand response programs [23] and to battery optimization through the application of the Elbow method to irradiance data [24].

However, comparisons of model order reduction methods, in terms of reduction ratio and reconstruction errors, have been limited to specific domains, in which specific methods are used. In mechanical systems [25], authors compared the Guyan reduction method, the Krylov subspace method and the balanced truncation method, and showed that the most suitable method depends on parameters of the problem to be solved (i.e. motion frequency). In structural dynamics [26], authors compared the modal displacement method, the balanced truncation method and the moment matching method, applied to a linear time-invariant dynamical system of an actuation frame, and showed that the modal displacement method captures best the global dynamics of the system. Nevertheless, a large variety of clustering methods can be employed to construct a reduced basis, such as Density Based Spatial Clustering of Applications with Noise (DBSCAN) [30], which works in a similar way than CVT, but only construct groups when a given density of vectors is found, excluding isolated ones. Clustering Using Representatives (CURE) [31] uses a different approach than CVT and DBSCAN, with the specificity of allowing initializing the groups of vectors using a sample of the data set. Eventually, Maximum Entropy Snapshot Sampling (MESS) [29] is another approach based on the information theory, as it works by computing the information gain associated with each vector of the data set, and constructs the reduced basis by extracting the vectors of highest information gain. Like SVD, these different methods enable to construct a reduced basis for model order reduction, in an engineering context. Therefore, an evaluation and comparison of their performances is required.

In this paper, we aim to carry out a comparative study between different snapshots selection methods: clustering techniques (CVT, DBSCAN, CURE), the MESS method, and SVD, which remains until now the reference for reduced basis construction. The methods are applied to computational physics, on a finite element model of an induction machine connected to the power network, which is a complex and realistic non-linear low frequency electromagnetic problem. We estimate the quality of reconstruction on the solution of the field distribution, but also on quantities of interest derived from it.

In Section 1, we introduce notations, the concept of a reduced basis, the snapshots selection methods and their implementation, as well as defining indicators. Section 2 presents the numerical experiment conducted to compare the methods, starting with the problem formulation. The comparison is made on field distribution but also on global quantities like energy and eddy current losses.

1. Methodologies

In this section, we will introduce the concept of reduced basis, as well as the different methods used to generate one from simulation results. Let consider T vectors x_i with N entries, solutions of a computational physic problem resulting from an analysis in the time domain of a parametrized problem. Considering a matrix $X \in \mathbb{R}^{N \times T}$ gathering the T vectors $x_i \in \mathbb{R}^N$ of the data set, we note $X_{*,c}$ the matrix containing the first c columns of X , and $X_{c,*}$ the matrix containing the first c rows of X . Then, we introduce the Frobenius norm of the matrix X , defined as follows:

$$\|X\|_F = \sqrt{\sum_{j=1}^N \sum_{i=1}^T x_{ij}^2} \quad (1)$$

1.1. Reduced basis

A reduced basis can be extracted from the matrix $X \in \mathbb{R}^{N \times T}$ using one of the methods presented hereafter. The idea is to construct a matrix $\Psi \in \mathbb{R}^{N \times m}$, with $m < T$, gathering orthonormal vectors $\varphi_i \in \mathbb{R}^N$, so that vectors of X can be expressed, or at least well approximated, as a linear combination of the vectors of Ψ . This expression, or projection, generates error, meaning that the basis Ψ does not enable to reconstruct exactly the matrix X but an approximation \tilde{X} . Doing so, we obtain a matrix $C \in \mathbb{R}^{m \times T}$ containing a set of coefficients $c_i \in \mathbb{R}^m$ defined as follows:

$$X \approx \tilde{X} = \Psi C \quad (2)$$

To obtain the coefficients c_i , we introduce the transpose of Ψ on both sides of (2), which gives us the projection of X in the reduced basis Ψ :

$$C = \Psi^t X \quad (3)$$

Then, instead of needing $N \times T$ terms to describe the matrix X , we only need $N \times m$ and $m \times T$ terms of the matrices Ψ and C respectively. We can see that if m is much lower than T , then the amount of data is much lower. We can later reconstruct the original matrix using (2), giving us the reconstructed matrix \tilde{X} , composed of T vectors $\tilde{x}_i \in \mathbb{R}^N$. The projection error can be computed for each vector of the matrix as follows:

$$\varepsilon_i = \|x_i - \tilde{x}_i\|_2 \quad (4)$$

This error is generated by the representation of the T vectors of matrix X with a linear combination of only m vectors and coefficients. Indeed, some information from one or multiple vectors of X may not be contained in any of the m vectors of Ψ . Furthermore, a low value for m will generate higher projection error and lower data needed to describe X , while a higher m gives a better quality of reconstruction but more data. A trade off should be found.

In the following, we present the different methods of snapshots selection, enabling to calculate the matrix Ψ , that we want to compare.

1.2. SVD

The Singular Value Decomposition [7] is a method for extracting the dominant components of a two-dimensional matrix via its singular values. It allows decomposing the matrix X into three matrixes U, Σ, V , defined as:

$$XV = U\Sigma \Leftrightarrow X = U\Sigma V^t \quad (5)$$

The columns of $U \in \mathbb{R}^{N \times N}$ are orthonormal eigenvectors of XX^t and are called the left singular vectors of X , while the columns of $V \in \mathbb{R}^{T \times T}$ are orthonormal eigenvectors of X^tX and are called the right singular vectors of X . The matrix $\Sigma \in \mathbb{R}^{N \times T}$ is a diagonal matrix composed of the singular values of X usually ranked in a decreasing order of magnitude. If $N > T$, which is often the case in computational engineering, only the first T terms of the diagonal are non-zeros, and we can rewrite equation (5) as:

$$X = U_{:T} \Sigma_{:T}^T V_{:T}^t \Leftrightarrow X = \sum_{k=1}^T u_k \lambda_k v_k^t \quad (6)$$

With $X_{:T}$ the matrix containing the first T columns of X , u_k and v_k respectively the k^{th} vector of the matrices U and V , and λ_k the k^{th} singular value. Given that expression, as well as the decreasing magnitude of the singular values, we can see that the contribution of the latest singular vectors will be very low in the reconstruction of X . This highlights the possibility of finding an approximation of X by a matrix X^m of rank $m < T$. In fact, the Eckart-Young theorem guarantees that the best low rank approximation of a matrix, in the perspective of the Frobenius norm, is given by its truncated SVD:

$$X^m = U_{:m} \Sigma_{:m}^m V_{:m}^t \quad (7)$$

From this approximation, we can extract the reduced basis Ψ_{SVD} as the matrix $U_{:m}$, which columns are orthogonal:

$$\Psi_{SVD} = U_{:m} \in \mathbb{R}^{N \times m} \quad (8)$$

1.3. CVT

The Centroidal Voronoï Tessellation (CVT) is a clustering method that can be used to select snapshots from the matrix X to construct a reduced basis. This is done by gathering the vectors x_i of X that look alike (which are close in the sense of the 2-norm) in groups (called Voronoï regions), and to represent each group by a single vector, which corresponds to the center of the Voronoï region [8].

The principle of CVT is to partition the discrete space generated by the set of vectors of X : $\Omega = \{x_i \in \mathbb{R}^N, i = 1, \dots, T\}$. It is therefore a matter of choosing a number of "regions" $V_j, j = 1, \dots, m$ to represent all vectors of X . Regions must satisfy the following conditions:

- Each region is included in Ω : $V_j \subset \Omega$
- Regions are disjoint: $V_j \cap V_h = \emptyset$ if $j \neq h$
- Regions cover Ω : $\bigcup_{j=1}^m V_j = \Omega$

The generators $\lambda_j \in \mathbb{R}^N, j = 1, \dots, m$ associated with the regions of Voronoï are defined as followed:

$$V_j = \left\{ x_k \in \Omega / \|x_k - \lambda_j\|_2 < \|x_k - \lambda_h\|_2 ; j = 1, \dots, m ; j \neq h ; k = 1, \dots, T \right\} \quad (9)$$

This means that each vector belongs to the Voronoï region corresponding to the generator to which it is closest.

Another characteristic of generators λ_j is that each of them must be merged with the center of mass of its associated region. This gives us a centered partition of space. In the case of a discrete domain of constant density, meaning that each vector has the same weight, the center of mass of a region is defined by, with $card(V_j)$ being the number of vectors in the region V_j :

$$Z_j = \frac{1}{card(V_j)} \sum_{x_k \in V_j} x_k \quad (10)$$

A CVT can be constructed by applying the Lloyd algorithm [27] on the matrix X . It is important to note that the first step of initialization of the generators can be critical in the construction of the CVT. The generators can be initialized by choosing randomly among the vector x_k , or from homogenous distribution of the x_k . The choice of the initial generators will condition the accuracy of the reduced

basis construction as we will see in the following. Furthermore, the merging of the centers of mass with the generators is an iterative process, so the user must define a precision criterion δ to ensure it. Its value should be low enough to ensure the quality of the partitioning, but not too small to avoid extra computational time.

The algorithm of CVT is presented in Figure 1. The inputs are the matrix $X \in \mathbb{R}^{N \times T}$, the number of regions m and the merging precision criterion δ , and the outputs are the m centers of mass Z_j .

Algorithm 1: CVT

Inputs:

the matrix $X \in \mathbb{R}^{N \times T}$

the number of regions m

the merging precision criterion δ

Outputs:

the m centers of mass Z_j

-
- 1) Initialization of m generators $\lambda_j \in \mathbb{R}^N, j = 1, \dots, m$
 - 2) Calculation of the regions V_j associated with these generators (9)
 - 3) Calculation of the center of mass Z_j associated with each region (10)
- As long as $\|Z_j - \lambda_j\|_2 > \delta, j = 1, \dots, m$:
- i) We take the centers of mass as new generators $\lambda_j = Z_j, j = 1, \dots, m$
 - ii) Calculation of the regions V_j associated with these generators (9)
 - iii) Calculation of the center of mass Z_j associated with each region (10)

End

Figure 1 – algorithm of the CVT

Once we obtain a CVT of the matrix X , we can construct the reduced basis Ψ_{CVT} by concatenating the obtained generators and applying a QR decomposition to orthogonalize the matrix:

$$\Psi_{CVT} = QR(\lambda_1, \dots, \lambda_m) \in \mathbb{R}^{N \times m} \quad (11)$$

We chose to use the QR decomposition as orthogonalization process in this work, because it is reliable and has a low computational cost [28].

1.4. MESS

Maximum Entropy Snapshot Sampling (MESS) is another snapshot selection method that can be used to derive a reduced basis from the matrix X . It has been proposed in [29]. It is based on the construction of a recurrence matrix, which measures the similarities between the different vectors, and on the notion of entropy, which corresponds for each vector to the quantity of information it contains.

The first step of this method is to calculate the recurrence matrix $R_\epsilon \in \mathbb{R}^{T \times T}$ corresponding to the matrix $X \in \mathbb{R}^{N \times T}$. To do this, we introduce a distance criterion ϵ , and the matrix is built as follows:

$$R_\epsilon^{i,g} = \begin{cases} 1 & \text{if } \|x_i - x_g\|_2 < \epsilon; \quad \epsilon > 0, i = 1, \dots, T, g = 1, \dots, T \\ 0 & \text{else} \end{cases} \quad (12)$$

The entry $R_\epsilon^{i,g}$ is equal to 1 if the two vectors x_i and x_g are close to each other. The idea is then to go through this recurrence matrix to deduce the entropy corresponding to each vector, and to determine those that contain the most “significant information”, which will be used to construct the reduced basis. For that, we define the ϵ -Frobenius potential v_ϵ , the ϵ -Frobenius entropy η_ϵ and the information gain h_ϵ :

$$v_{\epsilon_i} = v_{\epsilon}(X(:, 1:i)) = \frac{1}{i^2} \|R_{\epsilon}(1:i, 1:i)\|_F^2, i = 1, \dots, T \quad (13)$$

$$\eta_{\epsilon_i} = \eta_{\epsilon}(X(:, 1:i)) = -\log(v_{\epsilon_i}), i = 1, \dots, T \quad (14)$$

$$h_{\epsilon_i} = \eta_{\epsilon_{i+1}} - \eta_{\epsilon_i}, i = 1, \dots, T \quad (15)$$

We can then construct the reduced basis by extracting the m vectors $x_i^{\max(h_{\epsilon_i})}$ of X with the maximal information gain:

$$x_i^{\max(h_{\epsilon_i})} = \operatorname{argmax}(h_{\epsilon_i}) \quad (16)$$

The algorithm of MESS is presented in Figure 2. The inputs are the matrix $X \in \mathbb{R}^{N \times T}$, the number of vectors in the basis m and the distance criterion ϵ , and the outputs are the m vectors of maximum information gain $x_i^{\max(h_{\epsilon_i})}$.

Algorithm 2: MESS

Inputs:

the matrix $X \in \mathbb{R}^{N \times T}$
the number of bases m
the distance criterion ϵ

Outputs:

the m vectors of maximum information gain $x_i^{\max(h_{\epsilon_i})}$

1) Calculation of recurrence matrix (12)

For i from 1 to T :

- i) Calculation of the ϵ -Frobenius potential v_{ϵ_i} (13)
- ii) Calculation of the ϵ -Frobenius entropy η_{ϵ_i} (14)
- iii) Calculation of the information gain h_{ϵ_i} (15)

2) Selection of m vectors with maximum information gain $x_i^{\max(h_{\epsilon_i})}$ (16)

End

Figure 2 – algorithm of the MESS

The reduced basis is constructed by concatenation of m vectors with the highest value of h_{ϵ_i} , and an orthogonalization with QR decomposition:

$$\Psi_{MESS} = QR(x_1^{\max(h_{\epsilon_i})}, \dots, x_m^{\max(h_{\epsilon_i})}) \in \mathbb{R}^{N \times m} \quad (17)$$

1.5. DBSCAN

Density Based Spatial Clustering of Applications with Noise (DBSCAN) is a clustering method proposed in 1996 [30] and which uses the notion of density to create groups from the matrix X . With this method, one cannot define the number of groups that the algorithm will create, as it would be the case for CVT or MESS. However, the user will define a criterion associated to the notion of density, which will influence the creation of the groups of vectors. From the matrix X , we can define the density as the “ ϵ -neighborhood”:

$$N_{\epsilon}(x_i) = \{x_k \in X \mid \|x_i - x_k\|_2 < \epsilon\} \quad (18)$$

The first criterion set by the user is ϵ , which corresponds to a distance threshold. If the distance between two vectors x_i and x_k of X is lower than ϵ , these vectors will be gathered. The other criterion is \minPts , imposing a minimum number of vectors close together to create a group. From these criteria and the definition of the “ ϵ -neighborhood”, three categories of vectors appear:

- If the cardinal of $N_\epsilon(x_i)$ is greater than $minPts$, x_i is a “core” vector
- If the cardinal of $N_\epsilon(x_i)$ is between 0 and $minPts$, x_i is a “border” vector
- If the cardinal of $N_\epsilon(x_i)$ is 0, x_i is a “noise” vector

The three categories are illustrated in Figure 3, which displays the notion of density.

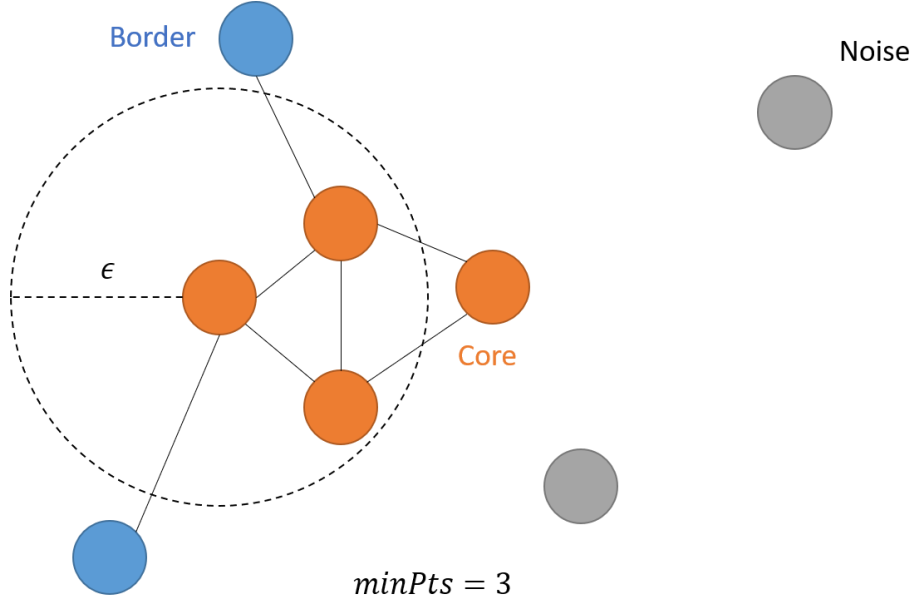


Figure 3 - DBSCAN scheme. The points in orange correspond to Core vectors, as we see 4 of them close to each other, which is superior to the value of $minPts$. The points in blue are Border vectors, being sufficiently close to only one other vector. The points in grey are Noise vectors, as they are too far to any other vector.

The DBSCAN algorithm consists in calculating the “ ϵ -neighborhood” associated with each vector of X to determine its category, and create groups of “core” and “border” vectors. It is presented in Figure 4. The inputs are the matrix $X \in \mathbb{R}^{N \times T}$, the distance threshold criterion ϵ and the minimum number of vectors to create a group $minPts$, and the outputs are the m centers of the created groups of vectors.

Algorithm 3: DBSCAN

Inputs:

the matrix $X \in \mathbb{R}^{N \times T}$

the distance threshold criterion ϵ

the minimum number of vectors to create a group $minPts$

Outputs:

the m centers K_i

While there are vectors x_i not ranked in any category (core, border, noise)

- i) Calculate “ ϵ -neighborhood”: $N_\epsilon(x_i)$ (18)
- ii) Categorize x_i by calculating $card(N_\epsilon(x_i))$ and comparing it to $minPts$
- iii) If x_i is a « core », create a cluster C_i gathering the vectors in $N_\epsilon(x_i)$
- iv) Categorize all the $x_k \subset N_\epsilon(x_i)$

End While

1) Include each “border” vector in its closest cluster

2) For each cluster C_i , calculate its center $K_i = \frac{1}{card(C_i)} \sum_{x_i \in C_i} x_i$

End

Figure 4 – algorithm of DBSCAN

The reduced basis is constructed by concatenation of the centers K_i , and orthogonalized with a QR decomposition.

$$\Psi_{DBSCAN} = QR(K_1, \dots, K_m) \in \mathbb{R}^{N \times m} \quad (19)$$

1.6. CURE

Clustering Using Representatives (CURE) is a hierarchical clustering algorithm proposed in 1998 [31]. It differs from the methods presented above, with the capacity of creating the clusters with a sample of the data instead of all of it, and representing each cluster by more than one representative vector. It allows to handle clusters with complex shapes and large database in a very efficient way. Representatives are calculated by taking n_{rep} furthest vectors in a cluster, and shrinking them toward the center of the cluster. The intensity of that shrinking can be chosen by the user, via a parameter θ .

With K_i the center of the cluster C_i , and x_i^{F1} one of its furthest vectors, we calculate the representative Rep_i with:

$$Rep_i^1 = x_i^{F1} + \theta(K_i - x_i^{F1}) \quad (20)$$

In this method, each vector in the sample data is initially considered as a cluster. The choice of the sample data is left to the user, and has an important impact on the quality of the reconstruction of X . Typically, it is required that the vectors of the sample data are already a good representation of the vectors of X . Then, clusters close to each other are fused until the desired number of clusters m is obtained. Finally, all the remaining vectors of X are compared to the clusters' representatives, and added in the closest cluster.

The algorithm of CURE is presented in Figure 5. The inputs are the matrix $X \in \mathbb{R}^{N \times T}$, the number of representative vectors for a group n_{rep} , the shrinking intensity parameter θ and the number of sample vectors $N_{Samples}$, and the outputs are the m centers of the created groups of vectors.

Algorithm 4: CURE

Inputs:

the matrix $X \in \mathbb{R}^{N \times T}$

the number of bases m

the number of representative vectors for a group n_{rep}

the shrinking intensity parameter θ

the number of sample vectors $N_{Samples}$

Outputs:

the m centers K_i

1) The first $N_{Samples}$ vectors x_i are considered as clusters C_i

While the number of clusters is superior than m

i) For each cluster C_i , calculate its center $K_i = \frac{1}{card(C_i)} \sum_{x_i \in C_i} x_i$

ii) For each cluster C_i , calculate its distance to the other clusters $D_i(k) = \|K_i - K_k\|_2$

iii) Fuse the closest clusters

End While

2) Calculate the n_{rep} representatives associated with each cluster C_i (20)

For j from $N_{Samples}$ to T :

i) For each cluster C_i , calculate the distance between its representatives and x_j

ii) Add x_j in its closest cluster

End for

3) For each cluster C_i , update its center $K_i = \frac{1}{card(C_i)} \sum_{x_i \in C_i} x_i$

Figure 5 – algorithm of CURE

Then, we can construct the reduced basis by concatenation of the centers K_i , and orthogonalize using a QR decomposition.

$$\Psi_{CURE} = QR(K_1, \dots, K_m) \in \mathbb{R}^{N \times m} \quad (21)$$

1.7. Projection error

In order to compare the efficiency of the presented methods for the selection of snapshots and the construction of reduced basis, we define the projection error. We calculate this error to evaluate the loss of information generated by the projection and reconstruction of the data (4). Then, the error relatively to the original data for each vector x_i is computed by:

$$\mathcal{E}_{x_i}^{Rel} = \frac{\varepsilon_i}{\|x_i\|_2} \quad (22)$$

2. Numerical experiments

In this section, we describe the experiments performed with the different methods presented above, in order to evaluate their ability to construct a reduced basis suited for the construction of a reduced order model from data provided by computational electromagnetics. Firstly, we present the mathematical formulation of the problem and simulation results, then we describe the computation of the reduced basis, and eventually we compare and analyze the results of the experiments in terms of projection error.

2.1. Problem formulation

We consider the application example of an induction machine, which can be described by magneto-quasistatic equations in the time domain.

$$\text{curl}(H) = J_{ind} + \sum_{j=1}^3 N_j i_j \quad (23)$$

$$\text{curl}(E) = -\frac{\partial B}{\partial t} \quad (24)$$

$$\text{div}(B) = 0 \quad (25)$$

$$\text{div}(J_{ind}) = 0 \text{ and } \text{div}(N_j) = 0 \quad (26)$$

Where B is the magnetic flux density, H the magnetic field, E the electric field, J_{ind} the eddy current density in the conducting domain, and N_j and i_j the unit current density vector and the current flowing through the winding j . The magnetic and electric behavior laws are defined by:

$$H = \vartheta(B)B \quad (27)$$

$$J_{ind} = \sigma E \quad (28)$$

With $\vartheta(B)$ the magnetic reluctivity, depending on B in the ferromagnetic material, and σ the electric conductivity, which is assumed to be constant in the conductive part and equal to zero in non-conductive parts (like in the windings of the stator, assumed to be stranded, and the lamination where the effect of eddy currents is negligible). We use the vector potential formulation [32] to solve the problem given by the previous equations. Introducing the magnetic vector potential A defined from (25) such as $B = \text{curl}(A)$, the electric field can be expressed from (24) by $E = -\frac{\partial A}{\partial t}$. Then, from (23) and (26), the equation to be solved is:

$$\text{curl}(\vartheta(B)\text{curl}(A)) + \sigma \frac{\partial A}{\partial t} = \sum_{j=1}^3 N_j i_j \quad (29)$$

with $A(t = 0) = 0$ in the conductive domain. We must also consider Dirichlet boundary conditions, to cancel the outgoing flux on the outer surface stator. The electromagnetic device is often connected to a voltage source, meaning that the voltage is imposed but not the currents i_j , which are then unknowns of the problem. The following relations allow to impose the voltage V_j at the terminals of the winding j , with R the resistance of windings, Φ_j the linkage magnetic flux and D_j the subdomain associated with the winding j :

$$\frac{d\Phi_j}{dt} + R i_j = V_j \quad (30)$$

$$\Phi_j = \int_{D_j} A \cdot N_j \, dx \quad (31)$$

The rotor is assumed to rotate at constant speed and the movement is considered by means of the time-stepping method [33]. In order to solve the problem, the fields A and N_j are discretized using respectively edge and facet elements, and we denote $X_A \in \mathbb{R}^N$ the vector of the components of A . The numerical solution of (29) is obtained using the Finite Element (FE) method. The full numerical model is then composed of a strong coupling between the FE model and the electric equation (30). Describing the FE model in the time domain, using the Euler implicit scheme, leads to solve an equation system, at each time step k , for $j = 1, 2, 3$:

$$[M^\vartheta + M(\theta_{k-1}) + M^\sigma \Delta t^{-1}] X_{A,k} - \sum_{j=1}^3 F_j i_{j,k} = M^\sigma \Delta t^{-1} X_{A,k-1} \quad (32)$$

$$F_j^t \Delta t^{-1} X_{A,k} + R i_{j,k} = V_{j,k} + F_j^t \Delta t^{-1} X_{A,k-1} \quad (33)$$

With Δt the time step, X^t the transpose of the matrix X , F_j the vector depending on N_j and $M(\theta_{k-1})$ the matrix used to consider the movement of the rotor.

2.2. Simulation of an induction machine

We consider a squirrel-cage induction machine similar to the one studied in [34], represented with a 2D-extruded mesh as shown in **Erreur ! Source du renvoi introuvable.** The mesh is composed of 86685 prism elements and 31360 hexahedral elements, the vector potential is discretized using edge functions. Due to the boundary conditions, the degrees of freedom associated with the edges on the boundary are equal to zero, and the unknowns of the problem are associated with the edges located inside the domain. The number of degrees of freedom is then $N = 76616$ unknowns. As said before, the machine is connected to a voltage supply delivering a nominal voltage $U_n = 6600 \, V$ at a frequency $f = 50 \, Hz$. We set the rotation speed to half the synchronous speed, giving $\Omega_{speed} = 743 \, r/min$. The eddy currents are neglected everywhere except in the bars, which means that we set $\sigma = 0$ everywhere but in the bars where $\sigma = 37,7 \cdot 10^6 \, Sm^{-1}$. We perform a simulation on $T = 2000$ time steps (≈ 15 electrical periods or $0.31s$), and thus obtain a result in the form of a matrix $X_A \in \mathbb{R}^{N \times T}$. We can then use this result to compute fields: the matrices of components of the magnetic flux density $X_B \in \mathbb{R}^{3N \times T}$ and of the density of eddy current $X_j \in \mathbb{R}^{3N \times T}$, which are of size $3N \times T$ because they gather 3 components per unknown; as well as global values: the eddy current losses $P_j \in \mathbb{R}^T$, and the magnetic energy $E_n \in \mathbb{R}^T$.

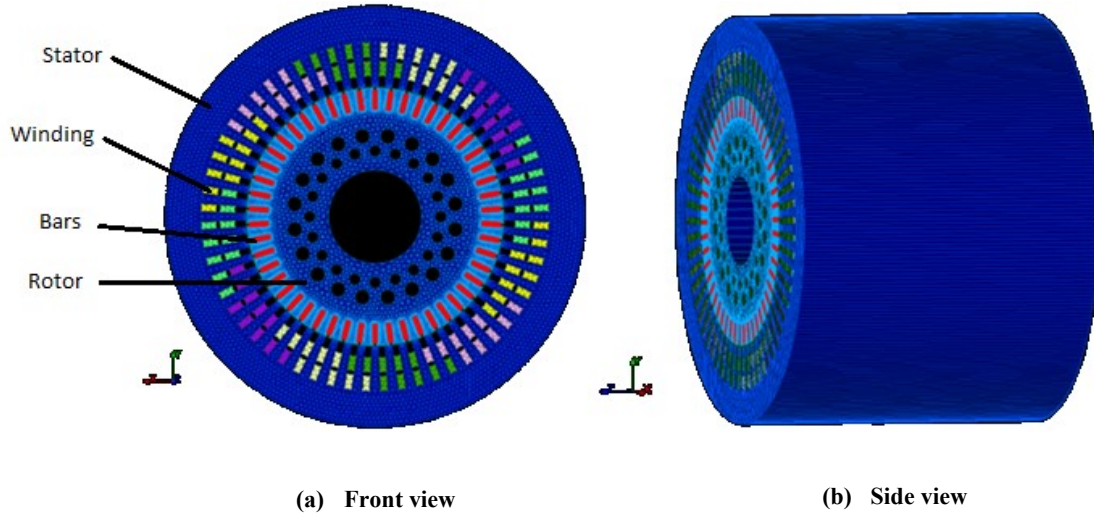


Figure 6 - mesh of the induction machine

Figure 8, Figure 9, Figure 10 and Figure 11 display fields and global values as results of a simulation of the induction machine. Figure 8 shows the magnetic flux density in the machine, in the transient (at 50th time step) and in steady-state (at 750th time step), and reveals a high magnetic saturation of the machine in the transient. Figure 9 shows the eddy current density distribution in the induction machine, also in transient (at 50th time step) and steady state (at 750th time step). We can see that the current density is only generated in the bars of the machine, which correspond to what we expect. Figure 10 and Figure 11 display the eddy current losses and the energy as global values of the induction machine. These graphs show clearly the presence of a transient and a steady state, with very high values of both losses and energy in the first electrical periods.

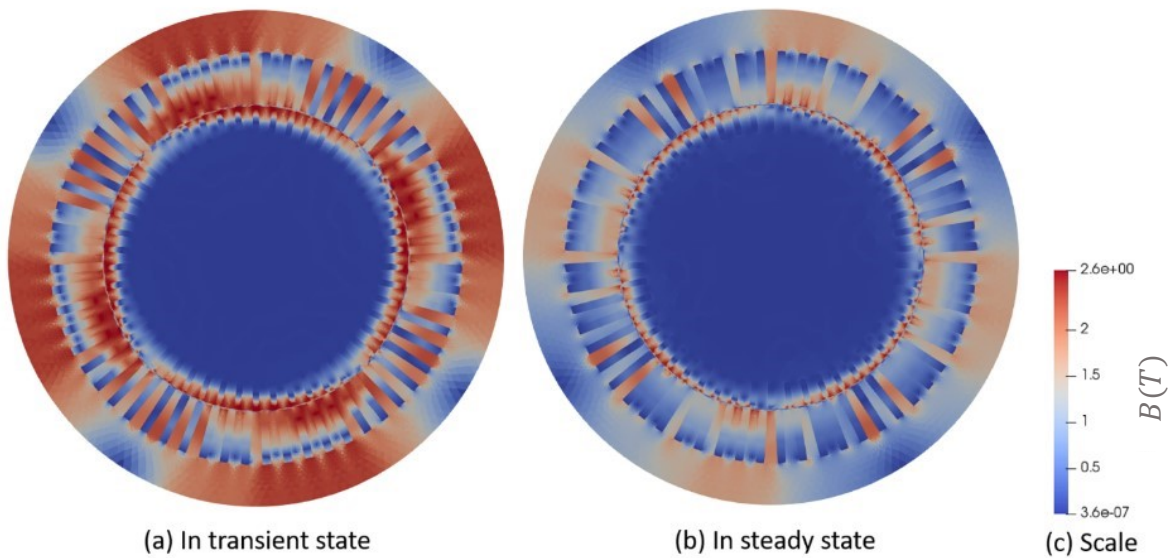


Figure 7 – electromagnetic field in the induction machine

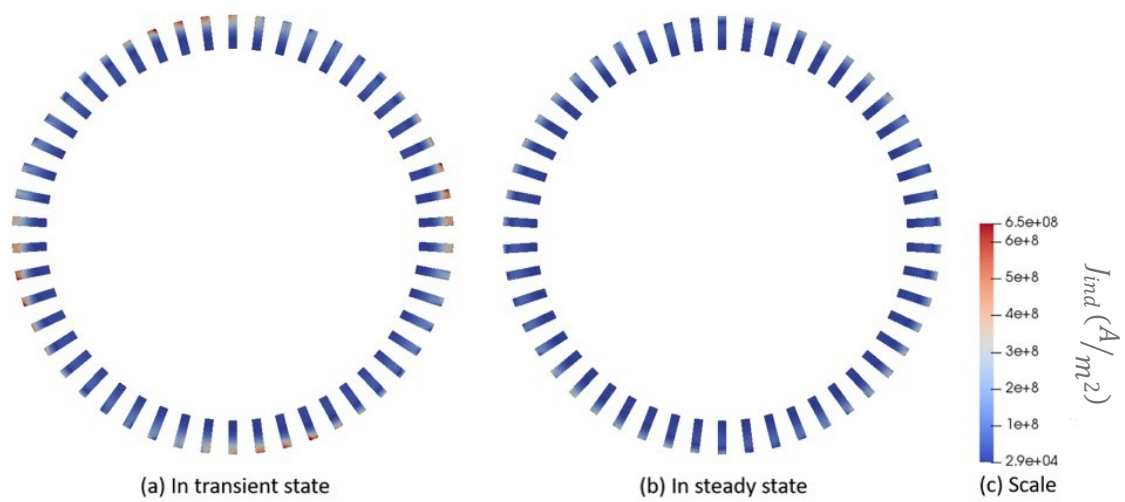


Figure 8 – density of eddy current in the induction machine

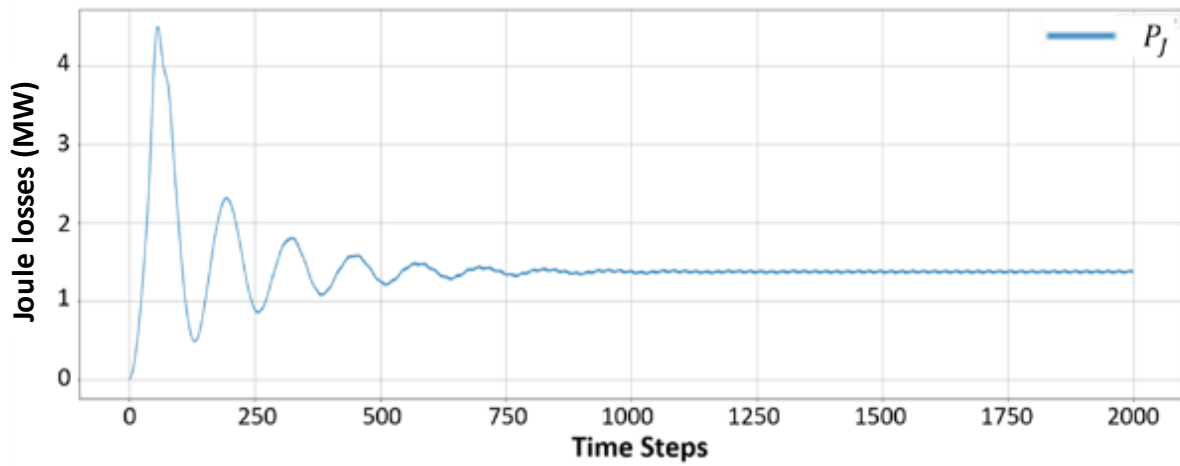


Figure 9 – Eddy current losses in the induction machine

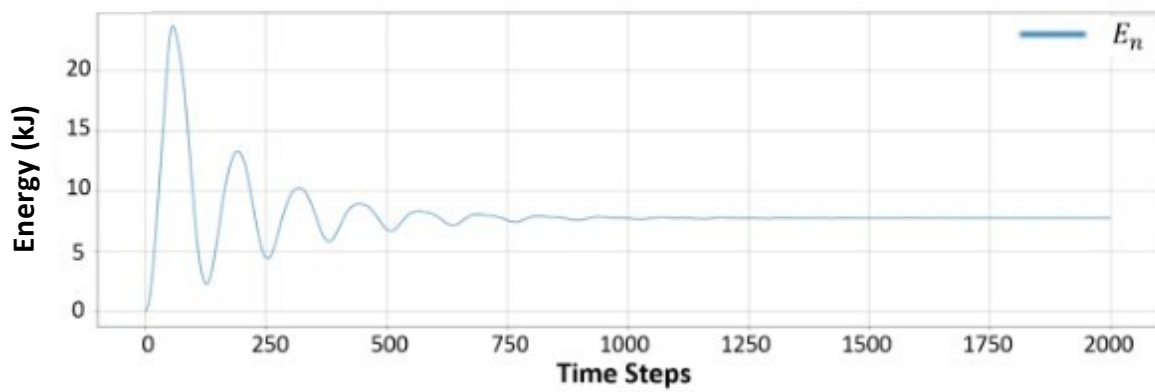


Figure 10 – energy in the induction machine

2.3. Computation of reduced basis

We compute a reduced basis from the matrix X_A , which gathers the T vectors of size N corresponding to the results of the simulation of the induction machine, with each method described above. In order to allow an objective comparison between the results obtained by the different reduced basis construction methods, we want all the basis to contain the same number of vectors m . Hence, we start by applying the DBSCAN on the matrix X_A , since it is the only method in which the user cannot impose the number of vectors in the reduced basis. We compute the density (18) setting $\epsilon = 0.03d_{max}$, where d_{max} is the distance (as a 2-norm) between the two furthest away vectors of X_A , and we set $minPts = 2$. We choose these values after a series of tests, in order to obtain a number of vectors $m = 212 \approx T/10$. This generates the reduced basis associated with the DBSCAN $\Psi_{DBSCAN} \in \mathbb{R}^{N \times m}$. Then, we can apply the other methods, setting the number of vectors in the reduced basis to m . The SVD method is computed on the matrix X_A and truncated at rank m (7), and we obtain the basis $\Psi_{SVD} \in \mathbb{R}^{N \times m}$. A CVT is performed on X_A , setting the number of regions to m . The generators chosen for the initialization are homogenously distributed vectors of X_A , which is more efficient than an initialization with a random choice. The merging precision criterion is set as $\delta = 0.0001$, and the algorithm generates the basis $\Psi_{CVT} \in \mathbb{R}^{N \times m}$. The CURE method is computed on the matrix X_A , with $n_{rep} = 2$, $\theta = 0.5$ and $N_{samples} = 360$, and gives us the reduced basis $\Psi_{CURE} \in \mathbb{R}^{N \times m}$. Eventually, the MESS is calculated on X_A , with $\epsilon = 0.002 * d_{max}$, and we obtain the reduced basis $\Psi_{MESS} \in \mathbb{R}^{N \times m}$. All values of the input parameters are listed in Table 1.

	DBSCAN	SVD	CVT	CURE	MESS
Inputs	$\epsilon = 0.03 * d_{max}$ $minPts = 2$	$m = 212$	$m = 212$ $\delta = 0.0001$	$m = 212$ $n_{rep} = 2$ $\theta = 0.5$ $N_{samples} = 360$	$m = 212$ $\epsilon = 0.002 * d_{max}$
Outputs	Ψ_{DBSCAN}	Ψ_{SVD}	Ψ_{CVT}	Ψ_{CURE}	Ψ_{MESS}

Table 1 – Parameters for the computation of reduced basis

We can then compare the offline CPU time of the different methods. However, SVD is performed using the compiled LAPACK library, while the other methods were programmed by us using Python programming language, and therefore are much less optimized. Nonetheless, CVT gives very interesting results, with an offline computation two times faster than SVD. DBSCAN yields an equivalent offline time than SVD, while CURE and MESS are around 3 times slower. We should recall that these two methods were programmed in Python and are not optimized. According to this study, we can't generalize the fact that the methods CURE and MESS are slower than SVD.

2.4. Projection and reconstruction of results

Once we have the reduced basis associated with each method, we perform the projection of the results matrix X_A in the reduced basis using (2). Then, we reconstruct the results by re-projecting the projected matrix in the reduced basis with (3). $C_A \in \mathbb{R}^{m \times T}$ is the projected matrix, $\Psi \in \mathbb{R}^{N \times m}$ the reduced basis used for projection and \tilde{X}_A the reconstructed results.

We perform the projection with each reduced basis computed beforehand, and obtain C_A^{SVD} , C_A^{CVT} , C_A^{DBSCAN} , C_A^{CURE} and C_A^{MESS} , all matrixes of size $\mathbb{R}^{m \times T}$.

We then reconstruct each projected matrix previously computed, using the associated reduced basis, and we obtain \tilde{X}_A^{SVD} , \tilde{X}_A^{CVT} , \tilde{X}_A^{DBSCAN} , \tilde{X}_A^{CURE} and \tilde{X}_A^{MESS} , all matrixes of size $\mathbb{R}^{N \times T}$.

In order to evaluate the efficiency of the methods to construct an efficient reduced basis to project the results, we firstly calculate a global projection error for each reconstructed matrix as follows:

$$\mathcal{E}_{X_A} = \|\tilde{X}_A - X_A\|_F \quad (34)$$

We obtain $\mathcal{E}_{X_A}^{SVD}$, $\mathcal{E}_{X_A}^{CVT}$, $\mathcal{E}_{X_A}^{DBSCAN}$, $\mathcal{E}_{X_A}^{CURE}$ and $\mathcal{E}_{X_A}^{MESS}$, which values are displayed in Figure 12. We can see that the value associated with the MESS is much higher than for other methods. Indeed, as mentioned in [35] where the MESS is compared to SVD, this method allows constructing the reduced basis much faster than SVD, and the basis constructed with MESS generates a better precision than with SVD for an identical calculation time. However, the MESS leads to a reduced basis of greater size than SVD for a given projection error. Then, as we evaluate the methods in term of reduction rate and precision, the MESS is not competitive with the other methods. From that observation, we propose to exclude the MESS from the rest of the study.

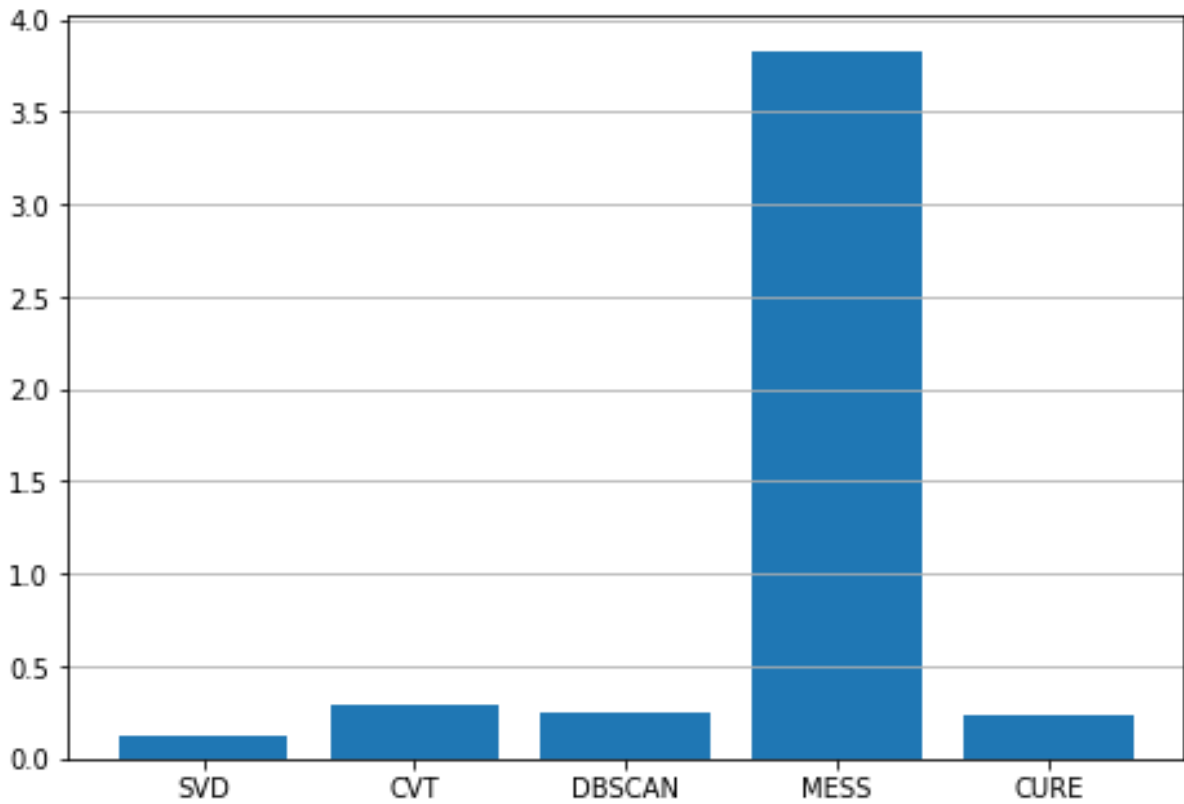


Figure 11 – Global reconstruction error for each method

We then compute relative error (22) associated with the reconstructed result for each method. We obtain $\mathcal{E}_{X_A}^{SVD}$, $\mathcal{E}_{X_A}^{CVT}$, $\mathcal{E}_{X_A}^{DBSCAN}$ and $\mathcal{E}_{X_A}^{CURE}$, all vectors of size \mathbb{R}^T . Figure 13 shows the value of the relative error for each method in function of the time steps. For all methods, the error calculated is lower than 0.01%, shows a bit of variability during the first time-steps and stays constant after the 500th. We can observe that the differences in the error generated by projection with the methods is concentrated during the first 500 time-steps, which correspond the transient state where we have the strongest variation of the field distribution. SVD gives the lowest error, in terms of a moving average, along all the time steps, as predicted by the Eckart-Young theorem. CVT generates a bigger error in the transient, but comes very close to the SVD as we get in the steady state. CURE allows reconstructing some results perfectly in the

first 250 time steps, which correspond to the samples used to initialize this method, and we can see the error being equal to zero. However, the reconstruction is degraded compared to SVD or CVT in steady state. Finally, DBSCAN gives similar results than CURE, with a slightly bigger error than CVT and SVD in the steady state, but a lower error than CVT in the transient.

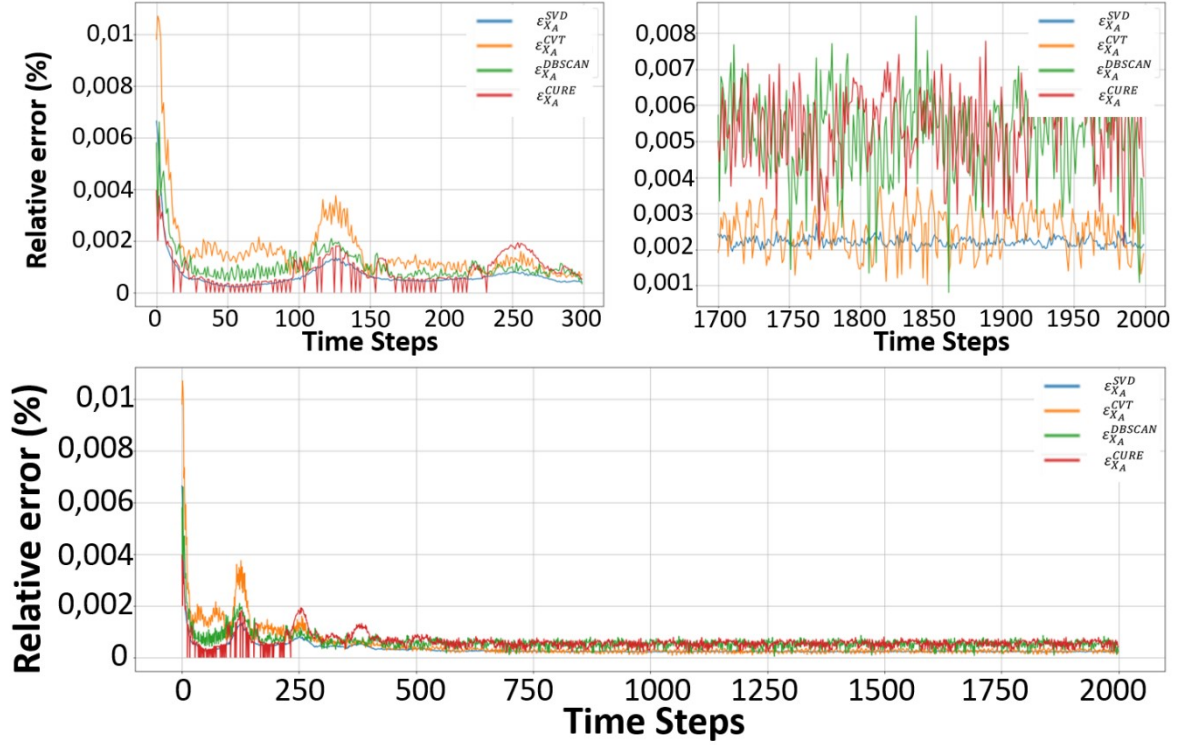


Figure 12 - Relative error of reconstructed results for SVD, CVT, DBSCAN and CURE (bottom) Zoom on transient state (top left) Zoom on steady state (top right)

2.5. Reconstruction of fields and global quantities

As presented above, we can compute fields and global quantities of the simulated machine from the matrix X_A . In order to further evaluate the quality of the reconstructed result, we use the matrixes \tilde{X}_A^{SVD} , \tilde{X}_A^{CVT} , \tilde{X}_A^{DBSCAN} and \tilde{X}_A^{CURE} to compute the fields and the global quantities, and compare them to those calculated with the result matrix X_A .

2.5.1. Magnetic flux density

At first, we compute the magnetic flux density B , corresponding to a spatial derivative of the vector potential A . We obtain the reconstructed magnetic flux density \tilde{X}_B^{SVD} , \tilde{X}_B^{CVT} , \tilde{X}_B^{DBSCAN} and \tilde{X}_B^{CURE} , all matrixes of size $\mathbb{R}^{3N \times T}$, and compute the relative errors $\varepsilon_{X_B}^{SVD}$, $\varepsilon_{X_B}^{CVT}$, $\varepsilon_{X_B}^{DBSCAN}$ and $\varepsilon_{X_B}^{CURE}$, vectors of size \mathbb{R}^T .

Figure 14 displays the relative error on the reconstructed magnetic flux density for each method. At first, we see a higher error generated than on the vector potential A . This can be explained by the important time and space variability of the field compared to the potential (induced by the space derivative), and by the construction of the reduced basis using snapshots of the vector potential and not those of the magnetic flux density. Then, we see that a strong variation of the error during the first time steps (around

30% for CVT), but it quickly decrease to 10% in transient state and less than 5% in steady state. DBSCAN and CURE yield lower error than CVT in transient state. The error oscillates around 2.5% in steady state for CURE and DBSCAN, and around 1.2% for SVD and CVT. SVD yields the lowest moving average error along all the time steps, and CVT, with peaks in the transient, generates a similar error than SVD in steady state. CURE generates a perfect reconstruction during the first time steps, as we saw beforehand, and DBSCAN give comparable results once again, with lower error than CVT in the transient but slightly higher in steady state.

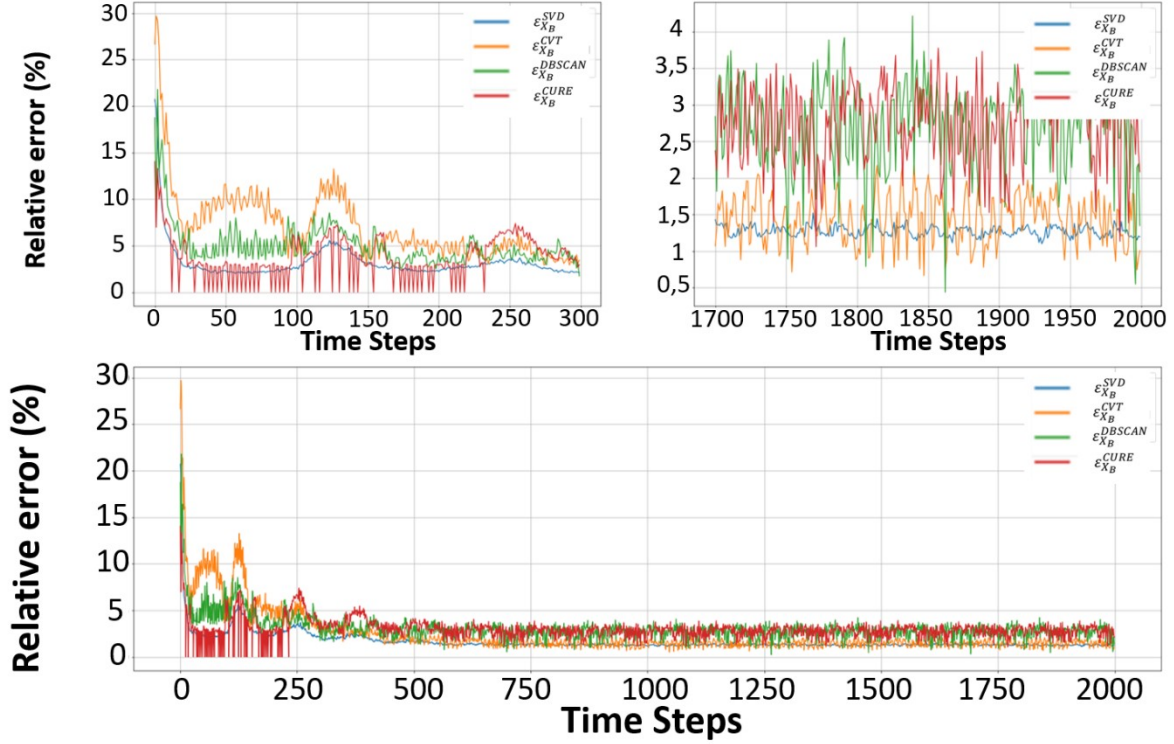


Figure 13 - Relative error of reconstructed electromagnetic fields for SVD, CVT, DBSCAN and CURE (bottom)
Zoom on transient state (top left) Zoom on steady state (top right)

2.5.2. Eddy current density

Similarly, we compute the eddy current density, corresponding to a time derivative of the vector potential in the conductive regions. We obtain the reconstructed density of eddy current \tilde{X}_J^{SVD} , \tilde{X}_J^{CVT} , \tilde{X}_J^{DBSCAN} and \tilde{X}_J^{CURE} , all matrixes of size $\mathbb{R}^{3N \times T}$, and compute the relative errors $\epsilon_{X_J}^{SVD}$, ϵ_J^{CVT} , ϵ_J^{DBSCAN} and $\epsilon_{X_J}^{CURE}$ vectors of size \mathbb{R}^T .

Figure 15 is a plot of the relative errors on the reconstructed current density distribution, and highlights again that the relative error associated with SVD is the lowest of all methods. CVT generates an error peaking at 20% in the transient, and decreasing in steady state to match SVD, around 2.5%. CURE does not allow a perfect reconstruction for some time steps as it was the case with the reconstruction of the magnetic flux density B (see 4.5.1). The current density corresponds to the time derivative of the vector potential, meaning that to have a perfect reconstruction of the current density requires having, at two consecutive time steps, a perfect reconstruction of the vector potential. DBSCAN and CURE give sensibly equivalent results, with a peak of error at 15% in the transient and a stabilization around 4% in steady state.

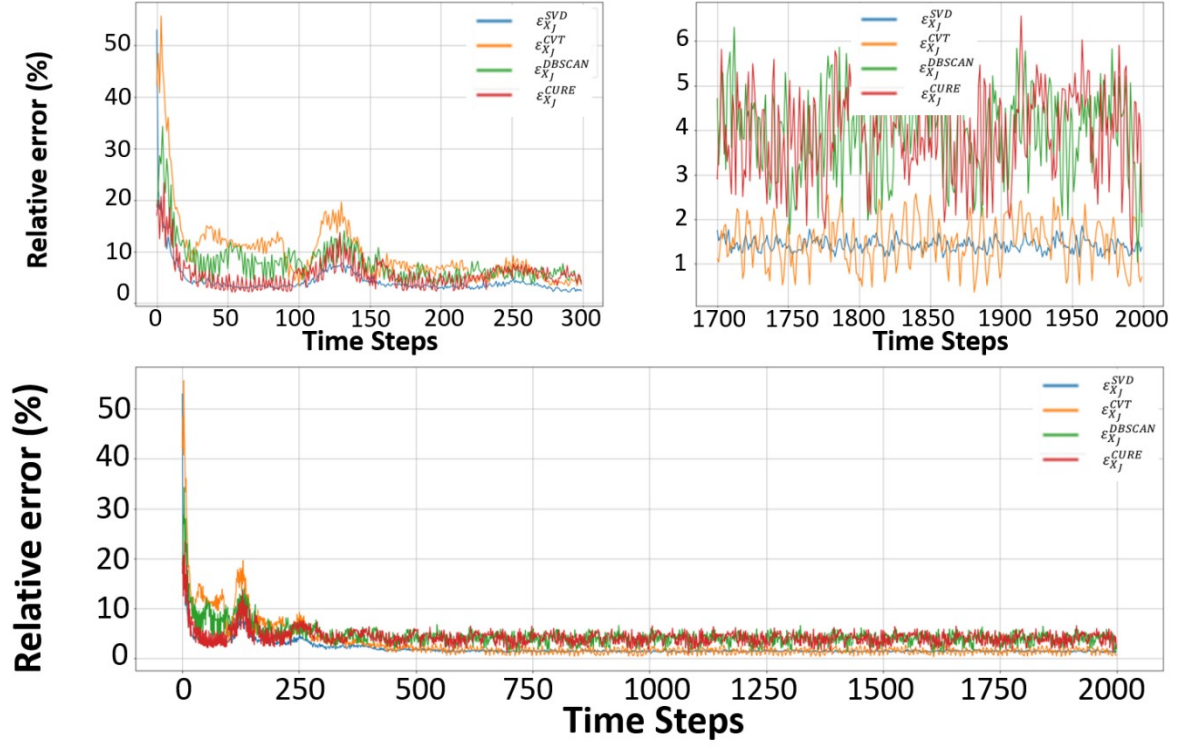


Figure 14 - Relative error of reconstructed density of current for SVD, CVT, DBSCAN and CURE (bottom) Zoom on transient state (top left) Zoom on steady state (top right)

2.5.3. Global quantities

We also reconstruct the eddy current losses \tilde{P}_J^{SVD} , \tilde{P}_J^{CVT} , \tilde{P}_J^{DBSCAN} and \tilde{P}_J^{CURE} , vectors of size \mathbb{R}^T , and calculate the relative errors $\varepsilon_{P_J}^{SVD}$, $\varepsilon_{P_J}^{CVT}$, $\varepsilon_{P_J}^{DBSCAN}$ and $\varepsilon_{P_J}^{CURE}$, vectors of size \mathbb{R}^T . Figure 16 displays the original and reconstructed eddy current losses time evolution, and shows that all methods allow a very good reconstruction of the losses.

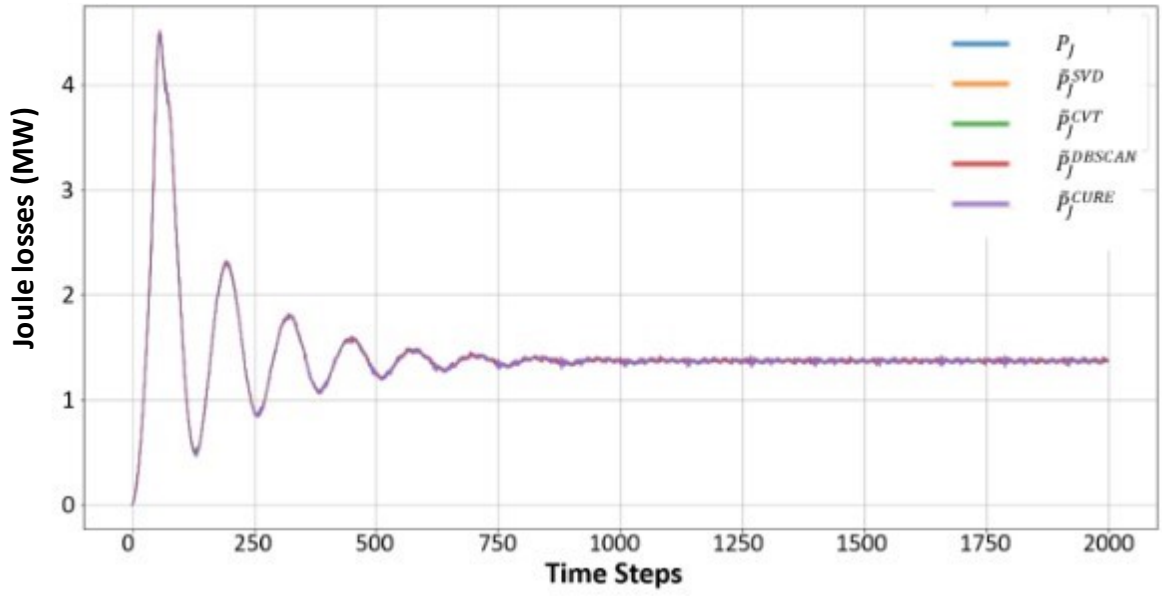


Figure 15 - Original eddy current losses and reconstructed with SVD, CVT, DBSCAN and CURE

Afterwards, we compute the reconstructed energy \tilde{E}_n^{SVD} , \tilde{E}_n^{CVT} , \tilde{E}_n^{DBSCAN} and \tilde{E}_n^{CURE} , vectors of size \mathbb{R}^T , and calculate the relative errors $\varepsilon_{E_n}^{SVD}$, $\varepsilon_{E_n}^{CVT}$, $\varepsilon_{E_n}^{DBSCAN}$ and $\varepsilon_{E_n}^{CURE}$, vectors of size \mathbb{R}^T . Figure 17 shows the original and reconstructed energies, and we see that all methods allow a very good reconstruction of the energy, with a slight deviation on the first peak.

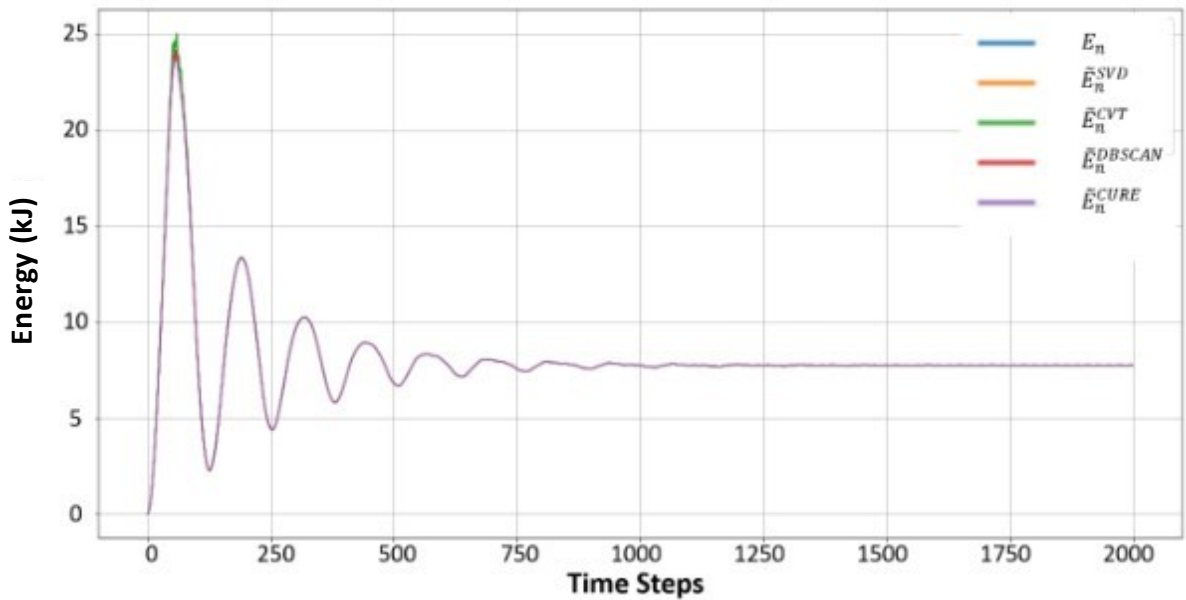


Figure 16 - Original Energy and reconstructed with SVD, CVT, DBSCAN and CURE

Figure 18 displays the relative errors associated with the reconstructed energies with each method, and shows that the error generated on the energy by the projection is low for all methods. CVT shows a peak of error around 6% in the transient, while DBSCAN rises to 2%. CURE and SVD generates a very low error even in the transient, below 1%. In steady state, we see that all the methods reconstruct the energy very well, with a relative error inferior to 1%.

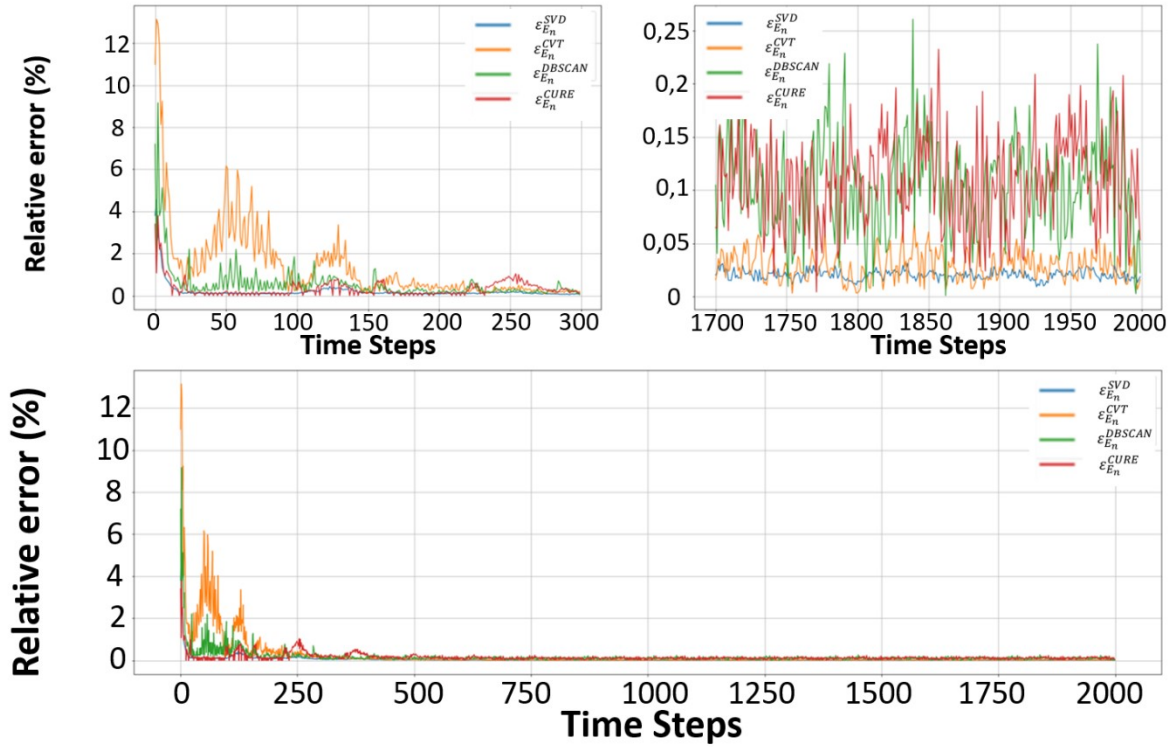


Figure 17 - Relative error of reconstructed Energy for SVD, CVT, DBSCAN and CURE (bottom) Zoom on transient state (top left) Zoom on steady state (top right)

To summarize these results, we calculate statistic values of the relative error, for each reconstructed value and each method. Figure 19 displays these values in the form of a boxplot, with the lowest tick corresponding to the minimum, the base of the box to the first quartile, the line in the box to the mean value, the top of the box to the third quartile and the highest tick to the maximum (scheme of the boxplot in the bottom-left box).

It highlights that projecting the result using SVD generates the lowest mean error, on every reconstructed value. In general, we see that CVT, CURE and DBSCAN give similar results, very close to those of SVD, with CURE giving slightly lower error.

These remarks corroborate what the different plots of reconstructed values, or of their relative errors, showed: SVD always generates the lowest deviation from the original value, as predicted by the Eckart-Young theorem. CVT is associated with a very precise reconstruction in steady-state (very close to SVD) but generates high peaks of error in transient. CURE and DBSCAN always give similar results, with a constant error along all time-steps, lower than CVT in transient but higher in steady-state.

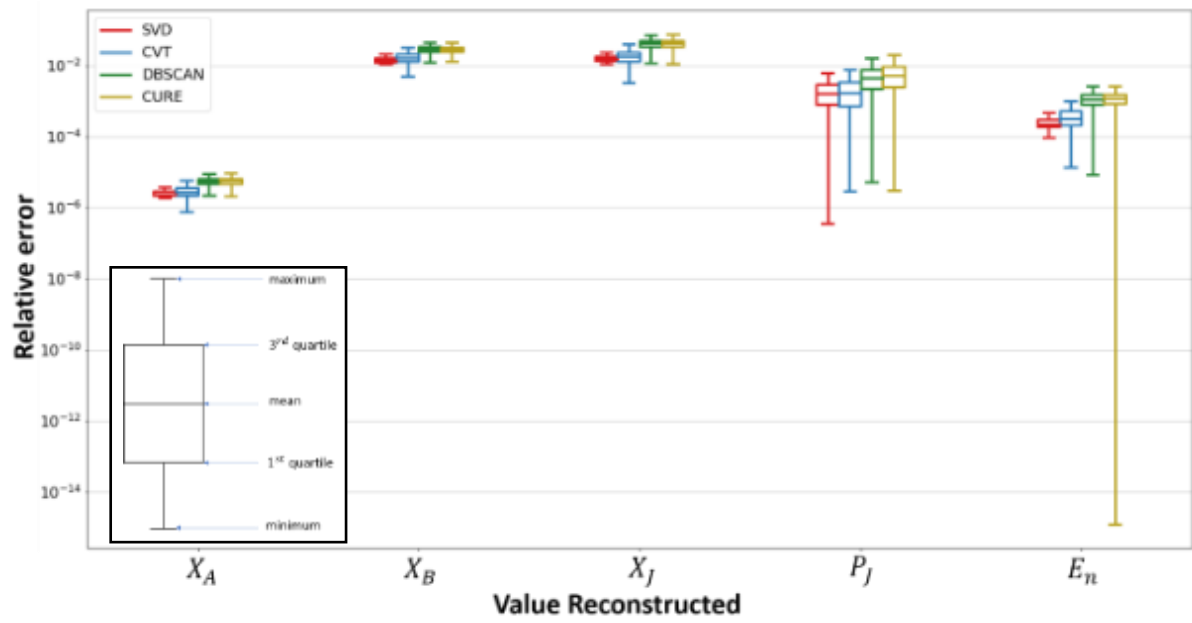


Figure 18 - Boxplot of statistic values of the relative error for each reconstructed value and each method

3. Conclusion

Different methods (SVD, CVT, MESS, CURE and DBSCAN) for constructing a reduced basis have been described and compared for simulation results projection on a realistic application in computational electromagnetics. The size of the constructed bases was fixed equally, and we investigated the influence of the chosen method on the projection error. To do so, we projected and reconstructed the results using the reduced bases, and the deviation to the original data generated by the operation was calculated, in term of relative error. We used this error to compare the quality of the reduced basis of each method and could see, as expected, that the SVD allows to project and reconstruct the results with the lowest deviation. We saw that MESS is not suited for the criterions of this comparison, as it is a method very efficient to obtain a reduced basis with a very low computational time, but which does not perform correctly in terms of precision when the target is the reduced basis size. We showed that CURE and DBSCAN give similar results, with an error being stable and close to SVD, and that CVT generates peaks of error during the transient state, but similar results to SVD in steady-state. For all methods, the reconstruction of the results was precise, with relative errors below 0.1%. We were then able to use the reconstructed results to compute fields and global values of the machine, and further analyze the quality of the reduced bases constructed. We showed that the deviations generated on both the electromagnetic field and the density of eddy current are similar, with relative errors around 10%. We also showed that eddy current losses and magnetic energy can be computed from the reconstructed results, and generate very little error ($\approx 5\%$). Moreover, it appeared that the projection error calculated from the unknowns of the problem (vector potential) gives a good image of the error on the other quantities, either local (magnetic field, eddy current density), or global (eddy current losses, magnetic energy). Consequently, we suggest that clustering methods, and particularly the CVT, can be valid alternatives to the SVD for reduced basis construction. Indeed, these methods showed similar results than the SVD in terms of projection error, and they may be used in cases where the SVD does not perform well, such as the construction of a basis from a very large set of preliminary simulation results. Besides generating reduced basis for model order reduction, the methods compared here are also valid for other applications, like the compression of results obtained from finite element simulation, to limit the storage space. In the application presented in this paper, the memory space can be divided by 10 without significant loss of accuracy. The memory space reduction can be controlled by means of the size of the reduced basis.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

- [1] A. Nouy. ‘Proper generalized decompositions for a priori model reduction of problems formulated in tensor product spaces: Alternative definitions and algorithms’. *In Proceedings of the Seventh International Conference on Engineering Computational Technology*, Civil-Comp Press, 2010.
- [2] A. Nouy. ‘A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations’. *Computer Methods in Applied Mechanics and Engineering*, 199(23) :1603–1626
- [3] G. Rozza, D. B. P. Huynh and A. T. Patera. ‘Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations’. *Archives of Computational Methods in Engineering*, 15(3) :229–275, 2008.
- [4] L. Montier, T. Henneron, S. Clenet, & B. Goursaud, (2021). “Model Order Reduction applied to a linear Finite Element model of a squirrel cage induction machine based on POD approach”, *IEEE Transactions on Magnetics*, 57(6), 1-4.
- [5] T. Delagnes, T. Henneron, S. Clenet, M. Fratila and J.-P. Ducreux, “Development of a FE Reduced Model on a Large Operating Range for a Squirrel Cage Induction Machine in Non Linear Case”, *CEFC 2022*, online
- [6] A. Saxena *et al.*, ‘A review of clustering techniques and developments’, *Neurocomputing*, vol. 267, pp. 664–681, Dec. 2017, doi: 10.1016/j.neucom.2017.06.053.
- [7] G. W. Stewart, ‘On the Early History of the Singular Value Decomposition’, *SIAM Rev.*, vol. 35, no. 4, pp. 551–566, Dec. 1993, doi: 10.1137/1035134.
- [8] Q. Du, V. Faber, et M. Gunzburger, « Centroidal Voronoi Tessellations: Applications and Algorithms », *SIAM Rev.*, vol. 41, no 4, p. 637-676, 1999.
- [9] O. Goury, C. Duriez. ‘Fast, generic and reliable control and simulation of soft robots using model order reduction’. *IEEE Transactions on Robotics*, IEEE, 2018, 34 (6), pp.1565 - 1576. [ff10.1109/TRO.2018.2861900ff](https://doi.org/10.1109/TRO.2018.2861900). [ffhal-01834483f](https://doi.org/10.1109/TRO.2018.2861900)
- [10] AT-M. Leung, R. Khazaka, ‘Parametric model order reduction technique for design optimization’. In : *2005 IEEE International Symposium on Circuits and Systems*. IEEE, 2005. p. 1290-1293.
- [11] X. Cheng, J. MA. Scherpen, ‘Clustering approach to model order reduction of power networks with distributed controllers’. *Advances in Computational Mathematics*, 2018, vol. 44, no 6, p. 1917-1939.
- [12] T. Lassila, A. Manzoni, A. Quarteroni et al. (2014) ‘Model Order Reduction in Fluid Dynamics: Challenges and Perspectives’. In: *Quarteroni, A. and Rozza, G., (eds.) Reduced Order Methods for Modeling and Computational Reduction*. MS&A Modeling, Simulation and Applications, 9 . , 235 - 273. ISBN 978-3-319-02089-1
- [13] A. Narasingam, P. Siddhamshetty, and J. Sang-Il Kwon, ‘Temporal clustering for order reduction of nonlinear parabolic PDE systems with time-dependent spatial domains: Application to a hydraulic fracturing process’. *AIChE Journal*, 2017, vol. 63, no 9, p. 3818-3831.
- [14] S. Jain, P. Tiso ‘Model order reduction for temperature-dependent nonlinear mechanical systems: a multiple scales approach’. *Journal of Sound and Vibration*, 2020, vol. 465, p. 115022.

- [15] G. Kerschen, J.-C. Golinval, A. Vakakis, *et al.* ‘The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: an overview’. *Nonlinear dynamics*, 2005, vol. 41, no 1, p. 147-169.
- [16] T. Henneron and S. Clénet. ‘Model order reduction of quasi-static problems based on POD and PGD approaches’. *The European Physical Journal Applied Physics*, 64(02) :24514, 2013.
- [17] Y. Sato and H. Igarashi. ‘Model reduction of three-dimensional eddy current problems based on the method of snapshots’. *Magnetics, IEEE Transactions on*, 49(5) :1697–1700, 2013.
- [18] D. Schmidhausler and M. Clemens. ‘Low-order electroquasistatic field simulations based on proper orthogonal decomposition’. *IEEE Transactions on Magnetics*, 48(2) :567–570, 2012.
- [19] R. Pulch, “Model order reduction and low-dimensional representations for random linear dynamical systems”, *Mathematics and Computers in Simulation*, Volume 144, 2018, Pages 1-20, ISSN 0378-4754
- [20] S. Torregrosa, V. Champaney, A. Ammar, V. Herbert, F. Chinesta, “Surrogate parametric metamodel based on Optimal Transport”, *Mathematics and Computers in Simulation*, Volume 194, 2022, Pages 36-63, ISSN 0378-4754
- [21] G. Deolmi, S. Müller, “A two-step model order reduction method to simulate a compressible flow over an extended rough surface”, *Mathematics and Computers in Simulation*, Volume 150, 2018, Pages 49-65, ISSN 0378-4754
- [22] H. Tertrais, R. Ibañez, A. Barasinski, Ch. Ghnatios, F. Chinesta, “On the Proper Generalized Decomposition applied to microwave processes involving multilayered components”, *Mathematics and Computers in Simulation*, Volume 156, 2019, Pages 347-363, ISSN 0378-4754
- [23] M. Mansoor, F. Grimaccia, S. Leva, M. Mussetta, “Comparison of echo state network and feed-forward neural networks in electrical load forecasting for demand response programs”, *Mathematics and Computers in Simulation*, Volume 184, 2021, Pages 282-293, ISSN 0378-4754
- [24] P. Ayuso, H. Beltran, J. Segarra-Tamarit, E. Pérez, “Optimized profitability of LFP and NMC Li-ion batteries in residential PV applications”, *Mathematics and Computers in Simulation*, Volume 183, 2021, Pages 97-115, ISSN 0378-4754
- [25] P. Koutsovasilis · M. Beitelschmidt. ‘Comparison of model reduction techniques for large mechanical systems A study on an elastic rod’, *Multibody Syst Dyn* (2008)
- [26] B. Besselink, U. Tabak, A. Lutowska, N. van de Wouw, H. Nijmeijer, et al.. ‘A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control’. *Journal of Sound and Vibration*, Elsevier, 2013, 332 (19), pp.4403-4422. [ff10.1016/j.jsv.2013.03.025](https://doi.org/10.1016/j.jsv.2013.03.025)ff.fhal-01711355f
- [27] S. Lloyd, ‘Least squares quantization in PCM’. *IEEE transactions on information theory*, 1982, vol. 28, no 2, p. 129-137.
- [28] K. Atkinson, An introduction to numerical analysis, *John wiley & sons*, 1991.
- [29] F. Kasolis and M. Clemens, ‘Maximum Entropy Snapshot Sampling for Reduced Basis Generation’, *ArXiv200501280 Cs Math*, May 2020. Available: <http://arxiv.org/abs/2005.01280>
- [30] M. Ester, H.-P. Kriegel, and X. Xu, ‘A density-based algorithm for discovering clusters in large spatial databases with noise’, *kdd*, vol. 96, no. 34, pp. 226-231, 1996.
- [31] S. Guha, R. Rastogi, and K. Shim, ‘CURE : An efficient clustering algorithm for large databases’, *ACM Sigmod record*, vol. 27, no 2, p. 73-84, 1998, doi: 10.1145/276305.276312

- [32] Y. Le Menach, ‘Contribution à la modélisation numérique tridimensionnelle des systèmes électrotechniques’, *Doctoral dissertation*, Lille 1, 1999.
- [33] T. W. Preston, A. B. J. Reece and PS Sangha. ‘Induction motor analysis by time-stepping techniques’, *IEEE Transactions on Magnetism*, vol. 24, no.1, pp. 471–474, 1988, doi: 10.1109/20.43959
- [34] J. Cheaytani, A. Benabou, A. Tounzi and M. Dessoude, ‘Stray Load Losses Analysis of Cage Induction Motor Using 3-D Finite-Element Method With External Circuit Coupling’, *IEEE Transactions on Magnetism*, vol. 53, no. 6, pp. 1-4, June 2017, doi: 10.1109/TMAG.2017.2661878
- [35] M. W. F. M. Bannenberg, F. Kasolis, M. Günther, and M. Clemens, ‘Maximum Entropy Snapshot Sampling for Reduced Basis Modelling’, *Preprint BUW-IMACM 20/46*, 2020