Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models - Application to transient elastodynamics in space-time domain

Lucas Boucinha, Amine Ammar, Anthony Gravouil, Anthony Nouy

Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models – Application to transient elastodynamics in space-time domain

L. Boucinha[a,*], A. Ammar[b], A. Gravouil[a,d], A. Nouy[c]

[a] *Université de Lyon, INSA-Lyon, LaMCoS, CNRS UMR5259, 18-20 rue des sciences, F-69621 Villeurbanne, France*
[b] *Arts et Métiers ParisTech, ENSAM Angers, 2 boulevard du Ronceray, F-49035 Angers, France*
[c] *LUNAM Université, Ecole Centrale Nantes, GeM UMR CNRS 6183, 1 rue de la Noë, BP 92101, 44 321 Nantes Cedex 3, France*
[d] *Institut Universitaire de France*

Abstract

It is now well established that separated representations built with the help of Proper Generalized Decomposition (PGD) can drastically reduce computational costs associated with solution of a wide variety of problems. However, it is still an open question to know if separated representations can be efficiently used to approximate solutions of hyperbolic evolution problems in space-time domain. In this paper, we numerically address this issue and concentrate on transient elastodynamic models. For such models, the operator associated with the space-time problem is non-symmetric and low-rank approximations are classically computed by minimizing the space-time residual in a natural L_2 sense, yet leading to non optimal approximations in usual solution norms. Therefore, a new algorithm has been recently introduced by one of the authors and allows to find a quasi-optimal low-rank approximation *a priori* with respect to a target norm. We presently extend this new algorithm to multi-field models. The proposed algorithm is applied to elastodynamics formulated over space-time domain with the time discontinuous Galerkin method in displacement and velocity. Numerical examples demonstrate convergence of the proposed algorithm and comparisons are made with classical *a posteriori* and *a priori* approaches.

Keywords: elastodynamics, separation of variables, Proper Orthogonal Decomposition (POD), Proper Generalized Decomposition (PGD), minimal residual, multi-field

1. Introduction

Standard discretization techniques, such as finite element method and time integration schemes, are now sufficiently well-established to be used for the solution of industrial problems. However, applying these methods in industrial context requires ever increasing computational resources. As an example, performing one simple transient elastodynamic simulation, on a structure discretized with one million nodes and over one million time steps, requires more than 7000 Gbytes in order to store a scalar solution over the space-time domain. Such a solution can never be stored in practice and engineers have the difficult task (before performing analyses) to select particular locations in space-time domain for partially post-processing the full solution, other calculated values being lost at the end of the simulation. Also, industrial problems usually depend on many parameters and engineers need to perform as many simulations as possible in optimization or uncertainty quantification contexts. Then, even if the impressive amount of computational resources today available potentially allows to deal with such models, one should propose innovative methods to better exploit these resources.

Reduced order models (ROMs) can be used to address this issue. They are usually based on projection of the full order model onto low dimensional reduced bases. The most popular reduced basis in elastodynamics is the truncated modal basis due to the physical interpretation of spatial modes [23, 37]. However, ROMs based on modal truncation are not efficient to simulate transient motions since eigen modes associated with a wide frequency band must be introduced in the reduced basis to get accurate results. More recently, ROMs based on proper orthogonal decomposition (POD) were proposed [7, 29, 28]. In this approach, the full order model in space is used to compute some solutions at particular time instants of the computational domain (called snapshots). Then, the idea is to build the reduced basis with the most dominant modes of the POD of the snapshots. This method is well-suited when reasonable comparability of model dynamics can be assumed [24]. Otherwise, many snapshots must be computed to obtain a reduced basis representative of all full order model states. POD method is called *a posteriori* since the reduced basis must be computed in a first step, based on partial knownledge of the full order model solution. In this paper, we present an *a priori* approach that allows finding a good approximation of the POD of the full space-time solution, without computing this full solution. It is based on a new algorithm recently proposed in [8].

Initially introduced in the 80's for the solution of non-linear mechanical models [30], ROMs based on PGD have been more

*Corresponding author : L. Boucinha, LaMCoS, INSA-LYON, 18-20 rue des Sciences, F-69621 Villeurbanne Cedex, France
Email address: lucas.boucinha@insa-lyon.fr (L. Boucinha)

recently generalized to high-dimensional problems [5, 1, 15, 35]. In the last decade, lots of contributions have been published, showing that many problems solutions can be well approximated using PGD of low ranks [16]. For evolution problems [2, 36], the aim is to approximate a space-time solution as a sum of products of space and time functions, that can be viewed as space-time modes. The number of space-time modes is called decomposition rank. The question is then, can we accurately approximate the solution of a given evolution problem with decomposition of low rank ? For second order hyperbolic problems the question is still open (see [10] for a first application of space-time PGD on the equation of waves motion). In this paper, we numerically address this issue and concentrate on solutions of transient elastodynamic models with two dimensions in space.

To evaluate if such solutions can be well approximated with separated representations of low rank, we first need to define what is the best approximation of a given rank M. Assuming the full solution is known, we should introduce a target norm in order to measure the error between the full solution and the low rank approximations. Then, the optimal approximation of rank M is defined as a minimizer of this error in the subset of rank-M tensors. This construction is called *a posteriori* decomposition [25] since one needs to calculate and be able to store the full solution before computing its low rank approximation. When this is not possible, a more challenging task is to compute a low rank approximation *a priori*, that is without knowing the full solution. This is aim of classical PGD methods. An even more challenging task is to compute the optimal low rank approximation *a priori* and this is aim of this paper.

For linear systems of equations with non-symmetric operators, as is the case of elastodynamic problems, PGD is classically computed by minimizing the discrete residual typically using (L_2-type) Euclidean norms [3, 36]. However, when operators are ill conditioned, the classical minimal residual PGD yields poorly accurate approximations with respect to usual solution norms. In elastodynamic problems, the number of space-time modes required to represent the full solution at a given accuracy is far from the optimal one. Therefore a new algorithm for non-symmetric problems has been introduced in [8] and allows finding *a priori* a quasi-optimal approximation of a given rank, with respect to a target metric. The idea is to introduce an ideal residual norm, so that minimizing this residual norm is equivalent to minimizing the error in the target norm. Notice that other variants have been also proposed in [36, 12].

We presently extend the algorithm proposed in [8] to multi-field models for its application in elastodynamics. This is motivated by the fact that recent time integration schemes used to compute an accurate solution of transient elastodynamic problems, are often based on multi-field formulations [27]. Here, we formulate the elastodynamic problem with two fields (displacement and velocity) and we use the Time Discontinuous Galerkin (TDG) method [26]. This formulation allows avoiding high frequency perturbations introduced by classical time schemes (such Newmark scheme) and that can increase the number of space-time modes required to approximate the full solution at a given accuracy [10]. Moreover, for elastodynamic models with two dimensions in space (or more), unknown solutions are vector fields (displacement and velocity vectors). In such case, each component of the unknown vectors is taken as a particular field.

This paper is structured as follows : in section 2, the optimal approximation of a given rank with respect to a target norm is defined and heuristic comparisons are made with approximations based on classical residual minimizations. In section 3, the ideal minimal residual-based approach allowing finding a quasi-optimal approximation with respect to the target metric is described. In section 4, we extend this approach to multi-field models and an example is given in section 5 for the elastodynamic problem with two dimensions in space. In section 6, space-time separated approximations of transient responses due to impact loads whose duration varies from 0.01 to 1 ms are computed. Numerical comparisons are made between *a posteriori* and *a priori* low-rank approximations, and convergence of the new algorithm is analyzed.

2. Optimality of low-rank approximations

In this section, we introduce space-time separated representations of solutions of single field models. First, we define the optimal separated approximation of a given rank with respect to a target metric. Then, we give some heuristic arguments to explain why this optimal decomposition may not be well approximated *a priori* with classical minimal residual approach if the operator associated with the considered model is ill conditioned.

2.1. A posteriori decomposition

Consider a scalar field $u(x,t)$ defined in a space-time domain $\Omega \times I$. After discretization on a given space-time mesh, the discrete representation of this field can be stored in a second order tensor denoted by $\mathbf{u} \in \mathcal{S} = \mathbb{R}^{N(S)} \otimes \mathbb{R}^{N(T)}$, where $N(S)$ and $N(T)$ give the size of the approximation spaces in the spatial and temporal dimensions respectively.

The aim of space-time decomposition is to find a low-rank approximation of a space-time field as a sum of products of spatial and temporal functions [30]. In the discrete case, it consists of approximating the tensor $\mathbf{u}$ as a sum of a few rank-one tensors. Each rank-one tensor can be viewed as a space-time mode. The number of space-time modes involved in the sum is called the decomposition rank M. Then, the rank-M space-time decomposition is denoted $\mathbf{u}_M$ and is searched in the subset $\mathcal{S}_M \subset \mathcal{S}$ defined as

$$\mathbf{u}_M \in \mathcal{S}_M = \left\{ \mathbf{u} \in \mathcal{S} \mid \mathbf{u} = \sum_{m=1}^{M} \underline{w}_m^S \otimes \underline{w}_m^T \right\} . \tag{1}$$

Since a space-time decomposition of fixed rank gives an approximation of the full tensor, we need a target norm in order to define the best approximation in a certain sense. Here, we choose the canonical norm for second order tensors, defined as

$$\left\| \mathbf{x} \right\|_2 = \sqrt{\mathbf{x} \overset{\mathrm{D}}{\cdot} \mathbf{x}} = \sqrt{\sum_{i,j} x_{ij}^2} ,$$

where the inner product "$\cdot^{\mathrm{D}}\cdot$" is the classical two times contracted product between second order tensors (that is $D = 2$).

Then, the optimal rank-M approximation $\mathbf{u}_M$ is searched as a minimizer of the distance to the solution $\mathbf{u}$, measured in the target norm, that is

$$\boxed{\mathbf{u}_M \in \arg\min_{\mathbf{u}^\star \in \mathcal{S}_M} \left\| \mathbf{u} - \mathbf{u}^\star \right\|_2^2}. \tag{2}$$

With the chosen target norm, problem (2) is simply the POD of rank-M of the full space-time solution. This construction is called *a posteriori* decomposition since it is necessary to calculate (and store) the full tensor $\mathbf{u}$ before to compute its low rank approximation. A more difficult task is to find an approximation of $\mathbf{u}$ while avoiding the calculation of $\mathbf{u}$. This is called *a priori* low rank approximation.

Remark 1. *In practice, the rank-M approximation $\mathbf{u}_M$, solution of problem (2), can be written*

$$\mathbf{u}_M = \sum_{m=1}^{M} \alpha_m \, \underline{w}_m^S \otimes \underline{w}_m^T \,, \tag{3}$$

with $\left\| \underline{w}_m^S \otimes \underline{w}_m^T \right\|_2 = 1$, for all $m = 1 \dots M$. The obtained decomposition is the Singular Value Decomposition (SVD) of the second order tensor $\mathbf{u}$ and it verifies the following property :

$$\left\| \mathbf{u} - \mathbf{u}_M \right\|_2^2 = \left\| \mathbf{u} \right\|_2^2 - \sum_{m=1}^{M} \alpha_m^2 \,. \tag{4}$$

2.2. A priori low rank approximation

A space-time separated approximation of the solution of a transient problem can be constructed *a priori* (that is without knowing $\mathbf{u}$) thanks to PGD methods [5, 3, 34]. After discretization, such problem can be written in tensor format as : find $\mathbf{u} \in \mathbb{R}^{N(S)} \otimes \mathbb{R}^{N(T)}$ such that

$$\begin{aligned} &\mathbf{A} \cdot^{\mathrm{D}} \mathbf{u} = \mathbf{b} \\ \text{with} \quad &\begin{cases} \mathbf{A} \in \mathbb{R}^{N(S)\times N(S)} \otimes \mathbb{R}^{N(T)\times N(T)} \\ \mathbf{b} \in \mathbb{R}^{N(S)} \otimes \mathbb{R}^{N(T)} \end{cases} \end{aligned} \tag{5}$$

where $\mathbf{A}$ is an invertible operator[1].

PGD methods can be efficiently applied only if $\mathbf{A}$ and $\mathbf{b}$ admit low rank representations[2], that is

$$\mathbf{A} = \sum_{m=1}^{M_A} \underline{\underline{A}}_m^S \otimes \underline{\underline{A}}_m^T \quad \text{and} \quad \mathbf{b} = \sum_{m=1}^{M_b} \underline{b}_m^S \otimes \underline{b}_m^T \,, \tag{6}$$

with moderate ranks M_A and M_b.

[1] In case of space-time problems, $\mathbf{A}$ is a fourth order tensor that can be identified with a linear operator that links a second order tensor with another second order tensor of the same size, that is $\mathbf{A} \cdot^{\mathrm{D}} \mathbf{u} = \mathbf{b} \Leftrightarrow \sum_{j_S=1}^{N(S)} \sum_{j_T=1}^{N(T)} A_{i_S j_S i_T j_T} u_{j_S j_T} = b_{i_S i_T}$.

[2] Indeed, assembly costs of linear operators involved in the alternated minimization process of PGD algorithms are mostly related to matrix-vector products and linearly depend on the ranks M_A and M_b.

Then, there are many ways to define PGD, the most used being the Galerkin-based PGD and the minimal residual-based PGD. However, for non-symmetric problems, convergence of the decomposition constructed with a Greedy rank-one algorithm has been proven only for minimal residual definition of the PGD (in a finite dimensional setting in [4] or infinite dimensional setting in [21]). Since operator $\mathbf{A}$ associated with elastodynamic problems is non-symmetric, we only consider the minimal residual definition of the PGD. Thus, the *a priori* low-rank approximation is searched among all rank-M tensors as a minimizer of the residual norm. Classically, residual minimization is performed using a canonical norm, that is

$$\boxed{\mathbf{u}_M \in \arg\min_{\mathbf{u}^\star \in \mathcal{S}_M} \left\| \mathbf{b} - \mathbf{A} \cdot^{\mathrm{D}} \mathbf{u}^\star \right\|_2^2}. \tag{7}$$

2.3. Optimality of minimal residual PGD

Minimal residual approximation (7) yields an optimal rank-M approximation in the sense that it minimizes the residual norm. However, our aim is to find a priori the optimal rank-M approximation (2) that minimizes the error defined with respect to the target norm we have chosen. Therefore, for a given rank M, one should ask if the minimal residual approximation (7) yields the best approximation (2). Here, we answer with some heuristic arguments. Defining the operator norm as $\left\| \mathbf{A} \right\|_2 = \max_{\substack{\mathbf{x}\in\mathcal{S} \\ \|\mathbf{x}\|_2=1}} \left\| \mathbf{A} \cdot^{\mathrm{D}} \mathbf{x} \right\|_2$, we have

$$\left\| \mathbf{A} \right\|_2^{-1} \left\| \mathbf{b} - \mathbf{A} \cdot^{\mathrm{D}} \mathbf{u}^\star \right\|_2 \leq \left\| \mathbf{u} - \mathbf{u}^\star \right\|_2 \leq \left\| \mathbf{A}^{-1} \right\|_2 \left\| \mathbf{b} - \mathbf{A} \cdot^{\mathrm{D}} \mathbf{u}^\star \right\|_2 . \tag{8}$$

Inequality (8) means that if we find a rank-M minimizer of the residual norm, it may yield a high error if the condition number $\kappa(\mathbf{A}) = \left\| \mathbf{A} \right\|_2 \left\| \mathbf{A}^{-1} \right\|_2$ is high. Then for ill-conditioned problem, minimal residual PGD could require the computation of more space-time modes than necessary to obtain an accurate enough approximation with respect to the target norm.

Remark 2. *From the computational point of view, similar algorithms can be used for a posteriori and a priori constructions of separated representations. To illustrate this point, we denote by $J(\mathbf{u})$ the functional involved in definitions (2) or (7). A first class of algorithms can be set up by minimizing the functional only with respect to the last mode in the current decomposition, other modes being supposed to be known. That is, we suppose $\mathbf{u}_{M-1}$ is known, and we look for the best rank-one enrichment $\mathbf{w} \in \mathcal{S}_1$ such that $J(\mathbf{u}_{M-1}+\mathbf{w})$ is minimum, then $\mathbf{u}_M = \mathbf{u}_{M-1}+\mathbf{w}$. This is called progressive or Greedy rank-one algorithms. For a posteriori decomposition, it yields the optimal rank-M approximation defined by the direct minimization of the functional on $\mathcal{S}_M$. For a priori decomposition, linear operator $\mathbf{A}$ also needs to be rank-one (that is $\mathbf{A} = \underline{\underline{A}}^1 \otimes \underline{\underline{A}}^2$). If this is not the case, progressive algorithms may converge to poorly accurate solutions [4, 20]. Then for $M_A > 1$, minimization must be performed with respect to all modes at each iteration (as done in problem (7)) if one wants to compute the optimal low-rank approximation. We call this approach direct algorithms. With this approach, the*

use of an alternating minimization algorithm requires the solutions of linear systems of size $MN(d)\times MN(d)$ (with $d = S$ or T) that cannot be solved for high values of the decomposition rank M. Then, different updating strategies are usually used in conjunction with progressive construction [35, 36, 22]. In this paper, we focus on direct and pure progressive constructions.

3. Algorithm based on ideal minimal residual formulation

In this section, we present an algorithm recently introduced in [8] that allows finding *a priori* a quasi-optimal low-rank approximation with respect to a target metric.

3.1. Ideal norm for residual minimization

The main idea is to introduce a residual norm whose minimization is equivalent to minimizing the error in the target metric. That is, we are looking for a symmetric definite positive operator $\mathbf{N}$ (defining the norm $\left\| \mathbf{x} \right\|_{\mathbf{N}} = \sqrt{\mathbf{x} \,\underline{\cdot}\, \mathbf{N} \,\underline{\cdot}\, \mathbf{x}}$) such that

$$\left\| \mathbf{u} - \mathbf{u}^\star \right\|_2 = \left\| \mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}^\star \right\|_{\mathbf{N}} . \tag{9}$$

This yields $\mathbf{N} = (\mathbf{A} \,\underline{\cdot}\, \mathbf{A}')^{-1}$ (where $\mathbf{A}'$ and $\mathbf{A}^{-1}$ denote transpose and inverse operations respectively). Then, the optimal rank-M decomposition (that is the POD of the space-time solution) can be defined as

$$\boxed{\mathbf{u}_M \in \arg \min_{\mathbf{u}^\star \in \mathcal{S}_M} \left\| \mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}^\star \right\|^2_{(\mathbf{A} \,\underline{\cdot}\, \mathbf{A}')^{-1}}} . \tag{10}$$

3.2. Direct construction with low-rank approximation of an auxiliary problem solution

Obviously, the norm involved in problem (10) is cumbersome from a computational point of view. To avoid its explicit calculation, we introduce an auxiliary variable $\mathbf{y} \in \mathcal{S}$, solution of the problem

$$(\mathbf{A} \,\underline{\cdot}\, \mathbf{A}') \,\underline{\cdot}\, \mathbf{y} = \mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}_M . \tag{11}$$

Introducing (11) in the stationary condition associated with problem (10), one can derive a gradient-type algorithm that can be viewed as an extension of algorithms introduced in [17, 14] to nonlinear approximation in $\mathcal{S}_M$. This algorithm reads at a given iteration $(k+1)$: knowing $\mathbf{u}_M^{(k)} \in \mathcal{S}_M$, find $\mathbf{u}_M^{(k+1)} \in \mathcal{S}_M$ such that

$$\mathbf{y}^{(k)} = (\mathbf{A} \,\underline{\cdot}\, \mathbf{A}')^{-1} \,\underline{\cdot}\, (\mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}_M^{(k)}) \tag{12a}$$

$$\mathbf{u}_M^{(k+1)} \in \arg \min_{\mathbf{u}^\star \in \mathcal{S}_M} \left\| \mathbf{u}_M^{(k)} + \mathbf{A}' \,\underline{\cdot}\, \mathbf{y}^{(k)} - \mathbf{u}^\star \right\|_2^2 \tag{12b}$$

Note that if equation (12a) is solved exactly, then this algorithm converges to the optimal rank-M approximation in one iteration whatever the initialization $\mathbf{u}_M^{(0)}$. However, this solution requires the same computational cost as the solution of equation (5). Therefore, it has been proposed in [8] to introduce a perturbation of this ideal approach. The idea is to compute a low-rank approximation of $(\mathbf{A} \,\underline{\cdot}\, \mathbf{A}')^{-1} \,\underline{\cdot}\, (\mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}_M^{(k)})$ that can be controlled with some precision. Then, convergence to a neighborhood of the optimal approximation can be proven depending on the precision of the low-rank approximation of the auxiliary variable [8]. Here, we use a slightly different approach. We propose to compute a low-rank approximation $\mathbf{y}_R \in \mathcal{S}_R$ of the auxiliary variable at fixed rank R. Since the auxiliary problem is symmetric, we compute this approximation with the progressive Galerkin PGD algorithm. Then, the direct algorithm for the *a priori* computation of a quasi-optimal low-rank approximation is called Ideal Minimal Residual PGD (denoted by (IMR)PGD in short) and reads : given $\mathbf{u}_M^{(k)} \in \mathcal{S}_M$, compute $\mathbf{u}_M^{(k+1)} \in \mathcal{S}_M$ such that

$$\mathbf{r}^{(k)} = \mathbf{b} - \mathbf{A} \,\underline{\cdot}\, \mathbf{u}_M^{(k)} , \tag{13a}$$

$$\mathbf{y}_R^{(k)} = \mathbf{(G)PGD}(\mathbf{A} \,\underline{\cdot}\, \mathbf{A}', \mathbf{r}^{(k)}, R) , \tag{13b}$$

$$\mathbf{u}_M^{(k+1)} \in \arg \min_{\mathbf{u}^\star \in \mathcal{S}_M} \left\| \mathbf{u}_M^{(k)} + \mathbf{A}' \,\underline{\cdot}\, \mathbf{y}_R^{(k)} - \mathbf{u}^\star \right\|_2^2 , \tag{13c}$$

where $\mathbf{(G)PGD}(\mathbf{A}, \mathbf{b}, M)$ gives a rank-M approximation of the solution of the linear system with operator $\mathbf{A}$ and right hand side $\mathbf{b}$, computed with a progressive Galerkin PGD algorithm [36].

Remark 3. *The rank-R approximation $\mathbf{y}_R$ of the solution $\mathbf{y}$ of the auxiliary problem (11) is the progressive Galerkin PGD of $\mathbf{y}$. In practice, it is computed under the form*

$$\mathbf{y}_R = \sum_{r=1}^{R} \beta_r \, \underline{y}_r^S \otimes \underline{y}_r^T \tag{14}$$

with $\left\| \underline{y}_r^S \otimes \underline{y}_r^T \right\|_{\mathbf{A} \,\underline{\cdot}\, \mathbf{A}'} = 1$, for all $r = 1 \ldots R$. Then, $\mathbf{y}_R$ verifies the following property [36] :

$$\left\| \mathbf{y} - \mathbf{y}_R \right\|^2_{\mathbf{A} \,\underline{\cdot}\, \mathbf{A}'} = \left\| \mathbf{y} \right\|^2_{\mathbf{A} \,\underline{\cdot}\, \mathbf{A}'} - \sum_{r=1}^{R} \beta_r^2 \tag{15}$$

Remark 4. *As explained in Remark 2, a direct algorithm for classical PGDs, using alternating minimization algorithm, may not be tractable since it requires alernated solutions of linear systems whose size linearly depends on the rank M. An important advantage of the proposed approach is that it does only require solutions of linear systems of size $N(d) \times N(d)$ (with $d = S$ or T) and thus, it can be used even for high values of decomposition rank M.*

Remark 5. *Computational costs associated with (IMR)PGD algorithm are mainly related to the progressive PGD algorithm used for the decomposition of the auxiliary problem's solution. Indeed, since $(\mathbf{u}_M^{(k)} + \mathbf{A}' \,\underline{\cdot}\, \mathbf{y}_R^{(k)})$ is already given in a low-rank format, the cost of step (13c) is negligible compared to the cost of step (13b).*

3.3. Convergence of the algorithm at fixed auxiliary rank R

Convergence of algorithm (13) towards a controlled neighborhood of the best rank-M approximation has been proven in [8] under the following assumptions:

- at each iteration (k), the rank-R approximation $\mathbf{y}_R^{(k)}$ of the solution $\mathbf{y}^{(k)}$ of the current auxiliary problem (12a) satisfies

$$\left\| \mathbf{y}^{(k)} - \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} \leq \delta \left\| \mathbf{y}^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} \quad \text{with} \quad 0 < \delta < \tfrac{1}{2} \,, \tag{16}$$

- and problem (13c) is solved with controlled precision.

In the present paper, $\mathbf{y}_R^{(k)}$ is the fixed rank-R progressive Galerkin PGD of $\mathbf{y}^{(k)}$, and then, thanks to property (15) and noting that $\left\| \mathbf{y}^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} = \left\| \mathbf{u} - \mathbf{u}_M^{(k)} \right\|_2$, we only ensure that

$$\left\| \mathbf{y}^{(k)} - \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} = \delta_R^{(k)} \left\| \mathbf{y}^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} \quad \text{with} \quad 0 \leq \delta_R^{(k)} < 1 \,, \tag{17}$$

where $\delta_R^{(k)} = \sqrt{1 - \frac{\sum_{r=1}^{R} (\beta_r^{(k)})^2}{\left\| \mathbf{u} - \mathbf{u}_M^{(k)} \right\|_2^2}}$. Therefore, an adaptive procedure allowing to find R such that $\delta_R^{(k)} \leq \delta$ should be defined in order to ensure convergence of the algorithm (see the approach used in [8]). However, we do not consider this aspect here since convergence of the algorithm has been numerically observed even for very small fixed rank R (typically $R = 1$, see section 6).

3.4. Error estimator

Thanks to property (17), we have

$$\tfrac{1}{1+\delta_R^{(k)}} \left\| \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} \leq \left\| \mathbf{u} - \mathbf{u}_M^{(k)} \right\|_2 \leq \tfrac{1}{1-\delta_R^{(k)}} \left\| \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'} . \tag{18}$$

Assuming that the auxiliary problem is solved with a reasonable precision, that means $\delta_R^{(k)}$ is small compared to 1, then $\left\| \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'}$ reflects the behavior of the true error $\left\| \mathbf{u} - \mathbf{u}_M^{(k)} \right\|_2$. A rigorous control of the algorithm would require the use of error estimation methods for the auxiliary problem in order to provide an upper bound of $\delta_R^{(k)}$. In practice, we here rely on a criterion based on the stagnation of $\left\| \mathbf{y}_R^{(k)} \right\|_{\mathbf{A} \,\underline{\mathrm{D}}\, \mathbf{A}'}$ in order to stop the algorithm.

4. Extension to multi-field models

In this section, we extend definitions and algorithms presented in sections 2 and 3 to multi-field models. We consider multi-field modeling of problems involving F scalar fields $u_1(x,t), \ldots, u_F(x,t)$ defined in a space-time domain $\Omega \times I$. These fields can be related to different physics (various examples can be found in [19, 32, 33, 6, 10]) or as illustrated in [9, 18], they can be different components of a vector field. Also, we allow different approximation spaces for each field, that is the discrete representation of a given field u_i is stored in a second order tensor $\mathbf{u}_i \in \mathbb{R}^{N(i,S)} \otimes \mathbb{R}^{N(i,T)}$ (where $N(i,d)$ gives the size of the approximation space related to the field i in the dimension $d = S$ or T).

4.1. Linear algebra on a product of tensor spaces

To shorten notations, we introduce the F-tuple $\{\mathbf{u}\} \in \mathsf{S}^F$, where S^F is a product of F tensor spaces. $\{\mathbf{u}\}$ can be viewed as a vector whose components are the F second order tensors $\mathbf{u}_i \in \mathbb{R}^{N(i,S)} \otimes \mathbb{R}^{N(i,T)}$. That is,

$$\{\mathbf{u}\} \in \mathsf{S}^F \Leftrightarrow \{\mathbf{u}\} \equiv \begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_F \end{bmatrix} . \tag{19}$$

Also, we introduce $\mathcal{L}(\mathsf{S}^F)$, the space of linear operators from S^F to S^F. An element $[\mathbf{A}] \in \mathcal{L}(\mathsf{S}^F)$ can be viewed as a matrix whose components are tensors $\mathbf{A}_{ij} \in \mathbb{R}^{N(i,S) \times N(j,S)} \otimes \mathbb{R}^{N(i,T) \times N(j,T)}$. That is,

$$[\mathbf{A}] \in \mathcal{L}(\mathsf{S}^F) \Leftrightarrow [\mathbf{A}] \equiv \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1F} \\ \vdots & & \vdots \\ \mathbf{A}_{F1} & \cdots & \mathbf{A}_{FF} \end{bmatrix} . \tag{20}$$

We introduce the following operations :

$$\{\mathbf{a}\} \,\underline{\mathrm{D}}\,. \{\mathbf{b}\} = \sum_{i=1}^{F} (\mathbf{a}_i \,\underline{\mathrm{D}}\, \mathbf{b}_i) \,,$$

$$[\mathbf{A}] \,\underline{\mathrm{D}}\,. \{\mathbf{a}\} = \{\mathbf{b}\} \Leftrightarrow \sum_{j=1}^{F} (\mathbf{A}_{ij} \,\underline{\mathrm{D}}\, \mathbf{a}_j) = \mathbf{b}_i \,,$$

where the additional point near the previously introduce D-times contracted product "$\,\underline{\mathrm{D}}\,$" is related to matrix-vector operations between components of the tuples. The first operation defines the natural inner product on S^F.

4.2. Optimal approximation with respect to the target norm

Following approaches used in [19, 32, 6, 9, 18, 10], we introduce a rank-M approximation[3] of each second order tensor $\mathbf{u}_i$. We denote this multi-field decomposition $\{\mathbf{u}\}_M$ and define the corresponding subset $\mathsf{S}_M^F \subset \mathsf{S}^F$ as

$$\mathsf{S}_M^F = \left\{ \{\mathbf{u}\} \in \mathsf{S}^F \mid \mathbf{u}_i = \sum_{m=1}^{M} \underline{w}_{im}^S \otimes \underline{w}_{im}^T \,,\; i = 1 \ldots F \right\} . \tag{21}$$

Then, we introduce a target norm in order to define the optimal decomposition. Here, we use the "canonical norm" of $F-$tuples whose components are second order tensors defined as

$$\left\| \{\mathbf{x}\} \right\|_2 = \sqrt{\{\mathbf{x}\} \,\underline{\mathrm{D}}\,. \{\mathbf{x}\}} \,. \tag{22}$$

Thus, for problems involving F second-order tensors, the optimal rank-M multi-field space-time decomposition with respect to the target norm is defined as

$$\boxed{\{\mathbf{u}\}_M \in \arg \min_{\{\mathbf{u}\}^\star \in \mathsf{S}_M^F} \left\| \{\mathbf{u}\} - \{\mathbf{u}\}^\star \right\|_2^2} . \tag{23}$$

This is the *a posteriori* decomposition of multi-field problems.

[3] A different rank M_i could be used for each tensor $\mathbf{u}_i$.

Remark 6. *Notice that with the chosen norm (22), optimal decomposition defined by (23) is equivalent to decomposition (2) applied to each field* $\mathbf{u}_i$*. However, when the* F*-tuple* $\{\mathbf{u}\}$ *contains different physical fields, the multi-field norm (22) involves summation of quantities that are not homogeneous in terms of physical units. Then, some weighting of the norm should be introduced in order to preserve physical homogeneity of the multi-field norm (see [10] for more details when the different fields are displacement and velocity fields). Notice that this weighting of the norms does not change the solution of problem (23).*

4.3. Classical Minimal residual PGD

As for single field case, we can also introduce *a priori* decompositions. We suppose the F tensors $\mathbf{u}_i$ are solutions of F coupled linear equations. These equations can be recast in a monolithic way as a unique block system. The multi-field linear system reads

$$[\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} \{\mathbf{u}\} = \{\mathbf{b}\} \Leftrightarrow \begin{bmatrix} \mathbf{A}_{11} & \cdots & \mathbf{A}_{1F} \\ \vdots & & \vdots \\ \mathbf{A}_{F1} & \cdots & \mathbf{A}_{FF} \end{bmatrix} \overset{\mathrm{D}}{\cdot\cdot} \begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_F \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_F \end{bmatrix}, \quad (24)$$

$$\text{with} \quad \begin{cases} \mathbf{A}_{ij} \in \mathbb{R}^{N(i,S)\times N(j,S)} \otimes \mathbb{R}^{N(i,T)\times N(j,T)} \\ \mathbf{b}_i \in \mathbb{R}^{N(i,S)} \otimes \mathbb{R}^{N(i,T)} \end{cases} \quad \forall i, j = 1 \dots F$$

where $[\mathbf{A}]$ is an invertible operator. Multi-field PGD can be efficiently applied when each block of the monolithic system is of low-rank. That is each of the components of the operator $[\mathbf{A}]$ and right hand side $\{\mathbf{b}\}$ admit low-rank representations of the form

$$\mathbf{A}_{ij} = \sum_{m=1}^{M_A(i,j)} \underline{\underline{A}}^S_{ijm} \otimes \underline{\underline{A}}^T_{ijm} \quad \text{and} \quad \mathbf{b}_i = \sum_{m=1}^{M_b(i)} \underline{b}^S_{im} \otimes \underline{b}^T_{im} \quad \forall i, j = 1 \dots F \quad (25)$$

with moderate ranks $M_A(i, j)$ and $M_b(i)$.

Then, following [10], we introduce a monolithic definition of *a priori* decomposition of multi-field problems. This approach differs from [19, 32, 6] in the sense that we do not partition the multi-field problem, thus preserving the better convergence properties of monolithic approaches while allowing different approximation spaces for each field. The decomposition is searched as a minimizer of the residual of the monolithic problem, among all decompositions in S^F_M, that is

$$\boxed{\{\mathbf{u}\}_M \in \arg \min_{\{\mathbf{u}\}^\star \in \mathsf{S}^F_M} \left\| \{\mathbf{b}\} - [\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} \{\mathbf{u}\}^\star \right\|_2^2}. \quad (26)$$

4.4. Ideal minimal residual approach

However, as for single field case, the quality of classical minimal residual-based decomposition depends on the condition number of the multi-field problem and do not give the optimal separated representation of rank M with respect to the target metric. Thus, an ideal minimal residual decomposition is introduced. In the multi-field case, this reads

$$\boxed{\{\mathbf{u}\}_M \in \arg \min_{\{\mathbf{u}\}^\star \in \mathsf{S}^F_M} \left\| \{\mathbf{b}\} - [\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} \{\mathbf{u}\}^\star \right\|^2_{([\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} [\mathbf{A}]')^{-1}}}. \quad (27)$$

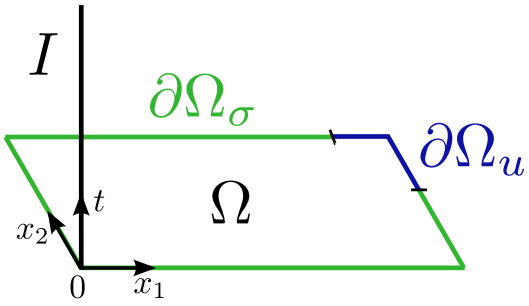

Figure 1: Geometry and boundary conditions.

Then, following section 3, we introduce an iterative scheme. Mimicking the direct construction introduced for single field case, we obtain the following iterative scheme in the multi-field case : given $\{\mathbf{u}\}^{(k)}_M \in \mathsf{S}^F_M$, compute $\{\mathbf{u}\}^{(k+1)}_M \in \mathsf{S}^F_M$ such that

$$\{\mathbf{r}\}^{(k)} = \{\mathbf{b}\} - [\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} \{\mathbf{u}\}^{(k)}_M, \quad (28a)$$

$$\{\mathbf{y}\}^{(k)}_R = \textbf{(G)PGD-MF}([\mathbf{A}] \overset{\mathrm{D}}{\cdot\cdot} [\mathbf{A}]', \{\mathbf{r}\}^{(k)}, R), \quad (28b)$$

$$\{\mathbf{u}\}^{(k+1)}_M = \arg \min_{\{\mathbf{u}\}^\star \in \mathsf{S}^F_M} \left\| \{\mathbf{u}\}^{(k)}_M + [\mathbf{A}]' \overset{\mathrm{D}}{\cdot\cdot} \{\mathbf{y}\}^{(k)}_R - \{\mathbf{u}\}^\star \right\|_2^2, \quad (28c)$$

where **(G)PGD-MF**$([\mathbf{A}], \{\mathbf{b}\}, M)$ gives a rank-M approximation of the solution of the multi-field linear system with operator $[\mathbf{A}]$ and right hand side $\{\mathbf{b}\}$, computed with a progressive Galerkin PGD as implemented in [10].

Remark 7. *Notice that in equation (28b), we use the same rank* R *for all auxiliary fields. A more adaptive strategy would consist in using a different rank* R_i *for each auxiliary field.*

5. Application to elastodynamic models in space-time domain

In this section, we illustrate construction of separated representations of operator $[\mathbf{A}]$ and right hand side $\{\mathbf{b}\}$ for elastodynamic models with two dimensions in space. A two-field (displacement $\underline{u}(\underline{x}, t)$ and velocity $\underline{v}(\underline{x}, t)$) weak formulation over space-time domain is first described. Then, separated representations of $[\mathbf{A}]$ and $\{\mathbf{b}\}$ are introduced considering a tensor structure of the form 2D-space × 1D-time (see Figure 1).

We assume small perturbations of material domain Ω over the time domain $I =]0, T[$ such that Ω does not evolve over I. Dirichlet boundary conditions (we consider imposed displacement $\underline{\bar{u}}$ and velocity $\underline{\bar{v}}$) are imposed on $\partial\Omega_u \times I$. Neumann conditions (we consider external load $\underline{p}$) are applied on $\partial\Omega_\sigma \times I$. Boundary is partitioned so that $\partial\overline{\Omega} = \partial\Omega_u \cup \partial\Omega_\sigma$ and $\partial\Omega_u \cap \partial\Omega_\sigma = \varnothing$. We suppose that $\partial\Omega_u$ and $\partial\Omega_\sigma$ do not evolve in time. Initial state is known and described by initial displacement $\underline{u}_0$ and velocity $\underline{v}_0$ defined on Ω.

Plain strain and linear elastic behavior of the medium are assumed. Strain and stress tensors ($\underline{\underline{\varepsilon}}$ and $\underline{\underline{\sigma}}$ respectively) are given by

$$\underline{\underline{\varepsilon}} = \tfrac{1}{2}(\underline{\nabla} \otimes \underline{u} + (\underline{\nabla} \otimes \underline{u})') \quad \text{and} \quad \underline{\underline{\sigma}} = \lambda \operatorname{tr}(\underline{\underline{\varepsilon}})\underline{\underline{I}} + 2\mu\, \underline{\underline{\varepsilon}}, \quad (29)$$

where tr($\underline{\underline{\varepsilon}}$) is the trace of $\underline{\underline{\varepsilon}}$, $\underline{\underline{I}}$ is the identity matrix, and λ and μ are Lamé's constants. We denote by e the thickness of the spatial domain.

5.1. Space-time weak formulation

We introduce the following function spaces :

$$\mathcal{U}(\underline{\bar{u}}) = \left\{\underline{u} : \Omega \times I \to \mathbb{R}^2 \mid \underline{u} = \underline{\bar{u}} \text{ on } \partial\Omega_u \times I + \text{regularity}\right\}$$
$$\mathcal{V}(\underline{\bar{v}}) = \left\{\underline{v} : \Omega \times I \to \mathbb{R}^2 \mid \underline{v} = \underline{\bar{v}} \text{ on } \partial\Omega_u \times I + \text{regularity}\right\}$$

and the following forms :

$$\mathsf{M}(\underline{u}\,,\,\underline{v}) = \int_\Omega \underline{u}\cdot\underline{v}\,\mathrm{d}\Omega, \tag{30}$$

$$\mathsf{K}(\underline{u}\,,\,\underline{v}) = \int_\Omega \underline{\underline{\sigma}}(\underline{u}) : \underline{\underline{\varepsilon}}(\underline{v})\,\mathrm{d}\Omega \quad (\text{with } \underline{\underline{\sigma}} : \underline{\underline{\varepsilon}} = \sum_{i,j=1}^{2}\sigma_{ij}\varepsilon_{ij}), \tag{31}$$

$$\mathsf{F}(\underline{v}) = \int_{\partial\Omega_\sigma} \underline{p}\cdot\underline{v}\,\mathrm{dS}\,. \tag{32}$$

The elastodynamic problem is weakly formulated over the space-time domain using the two-field Time Discontinuous Galerkin (TDG) method, as proposed in [26]. Thus, the space-time domain is decomposed into subintervals called space-time slabs : $\Omega\times I = \bigcup_{i=1}^{N_{slab}}\Omega\times I_i$ with $I_i =]t_{i-1}^+, t_i^-[$ where $t_0^+ = 0$ and $t_{N_{slab}}^- = T$. Then, the weak formulation reads : find $\underline{u}\in\mathcal{U}(\underline{\bar{u}})$ and $\underline{v}\in\mathcal{V}(\underline{\bar{v}})$ such that

$$\begin{aligned}\mathsf{A}(\{\underline{u},\underline{v}\},\{\underline{u}^\star,\underline{v}^\star\}) = \mathsf{b}(\{\underline{u}^\star,\underline{v}^\star\})\\ \forall \underline{u}^\star\in\mathcal{U}(\underline{0}), \forall \underline{v}^\star\in\mathcal{V}(\underline{0}),\end{aligned} \tag{33}$$

where the space-time bilinear form A is given by

$$\begin{aligned}\mathsf{A}(\{\underline{u},\underline{v}\},\{\underline{u}^\star,\underline{v}^\star\}) &= \sum_{i=1}^{N_{\text{slab}}}\int_{I_i}\mathsf{M}(\rho\,\dot{\underline{v}}\,,\,\underline{v}^\star) + \mathsf{K}(\underline{u}\,,\,\underline{v}^\star)\,\mathrm{dt}\\
&+\sum_{i=1}^{N_{\text{slab}}}\int_{I_i}\mathsf{K}(\dot{\underline{u}}-\underline{v}\,,\,\underline{u}^\star)\,\mathrm{dt}\\
&+\sum_{i=2}^{N_{\text{slab}}}\mathsf{M}(\rho\,\underline{v}(\underline{x},t_{i-1}^+)-\rho\,\underline{v}(\underline{x},t_{i-1}^-)\,,\,\underline{v}^\star(\underline{x},t_{i-1}^+))\\
&+\sum_{i=2}^{N_{\text{slab}}}\mathsf{K}(\underline{u}(\underline{x},t_{i-1}^+)-\underline{u}(\underline{x},t_{i-1}^-)\,,\,\underline{u}^\star(\underline{x},t_{i-1}^+))\\
&+\mathsf{M}(\rho\,\underline{v}(\underline{x},t_0^+)\,,\,\underline{v}^\star(\underline{x},t_0^+)) + \mathsf{K}(\underline{u}(\underline{x},t_0^+)\,,\,\underline{u}^\star(\underline{x},t_0^+))\,,\end{aligned} \tag{34}$$

and the linear operator b is given by

$$\mathsf{b}(\{\underline{u}^\star,\underline{v}^\star\}) = \sum_{i=1}^{N_{\text{slab}}}\int_{I_i}\mathsf{F}(\underline{v}^\star)\,\mathrm{dt} + \mathsf{M}(\rho\,\underline{v}_0\,,\,\underline{v}^\star(\underline{x},t_0^+)) + \mathsf{K}(\underline{u}_0\,,\,\underline{u}^\star(\underline{x},t_0^+))\,. \tag{35}$$

5.2. 2D-Space × 1D-Time decomposition

We give here an illustration of separated representation of bilinear and linear forms involved in (34) using a 2D-space × 1D-time decomposition. In this case, there is no restriction on the geometry of the spatial domain, nor on the supports of Dirichlet boundary conditions.

All components of the displacement and velocity fields are considered as independent fields, that is

$$\underline{u}(\underline{x},t) = \sum_{i=1}^{2} u_i(\underline{x},t)\underline{e}_i \quad\text{and}\quad \underline{v}(\underline{x},t) = \sum_{i=1}^{2} v_i(\underline{x},t)\underline{e}_i\,. \tag{36}$$

To simplify, we discretize each component of the displacement and velocity fields with the same approximation space. That is for $i = 1$ or 2,

$$u_i(\underline{x},t) = \underline{\Phi}(\underline{x})\otimes\underline{\Psi}(t)\overset{\mathrm{D}}{\cdot}\mathbf{u}_i + \bar{u}_i(\underline{x},t)\,, \tag{37}$$
$$v_i(\underline{x},t) = \underline{\Phi}(\underline{x})\otimes\underline{\Psi}(t)\overset{\mathrm{D}}{\cdot}\mathbf{v}_i + \bar{v}_i(\underline{x},t)\,, \tag{38}$$

where $\underline{\Phi}(\underline{x})$ are continuous finite elements basis functions defined on Ω (and vanishing on $\partial\Omega_u$) and $\underline{\Psi}(t)$ are piecewise continuous finite elements basis functions defined on I. Virtual fields are discretized similarly for $i = 1$ or 2,

$$u_i^\star(\underline{x},t) = \underline{\Phi}(\underline{x})\otimes\underline{\Psi}(t)\overset{\mathrm{D}}{\cdot}\mathbf{u}_i^\star\,, \tag{39}$$
$$v_i^\star(\underline{x},t) = \underline{\Phi}(\underline{x})\otimes\underline{\Psi}(t)\overset{\mathrm{D}}{\cdot}\mathbf{v}_i^\star\,. \tag{40}$$

Then, introducing approximations (37) to (40) in the weak form (34), we obtain the discrete problem in tensor format : find $\mathbf{u}_1\in\mathbb{R}^{N(S)}\otimes\mathbb{R}^{N(T)}$, $\mathbf{u}_2\in\mathbb{R}^{N(S)}\otimes\mathbb{R}^{N(T)}$, $\mathbf{v}_1\in\mathbb{R}^{N(S)}\otimes\mathbb{R}^{N(T)}$ and $\mathbf{v}_2\in\mathbb{R}^{N(S)}\otimes\mathbb{R}^{N(T)}$ such that

$$\begin{bmatrix}\underline{\underline{K}}_{11}\otimes\underline{\underline{A}} & \underline{\underline{K}}_{12}\otimes\underline{\underline{A}} & -\underline{\underline{K}}_{11}\otimes\underline{\underline{B}} & -\underline{\underline{K}}_{12}\otimes\underline{\underline{B}}\\
\underline{\underline{K}'}_{12}\otimes\underline{\underline{A}} & \underline{\underline{K}}_{22}\otimes\underline{\underline{A}} & -\underline{\underline{K}'}_{12}\otimes\underline{\underline{B}} & -\underline{\underline{K}}_{22}\otimes\underline{\underline{B}}\\
\underline{\underline{K}}_{11}\otimes\underline{\underline{B}} & \underline{\underline{K}}_{12}\otimes\underline{\underline{B}} & \underline{\underline{M}}_{11}\otimes\underline{\underline{A}} & \mathbf{0}\\
\underline{\underline{K}'}_{12}\otimes\underline{\underline{B}} & \underline{\underline{K}}_{22}\otimes\underline{\underline{B}} & \mathbf{0} & \underline{\underline{M}}_{22}\otimes\underline{\underline{A}}\end{bmatrix}\overset{\mathrm{D}}{\cdot}\begin{bmatrix}\mathbf{u}_1\\ \mathbf{u}_2\\ \mathbf{v}_1\\ \mathbf{v}_2\end{bmatrix} = RHS\,, \tag{41}$$

where operators defined on the spatial domain are classical stiffness and mass matrices given by

$$\underline{\underline{K}}_{11} = \int_\Omega e\left((\lambda+2\mu)\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_1}\otimes\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_1} + \mu\,\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_2}\otimes\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_2}\right)\mathrm{d}x_1\mathrm{d}x_2\,,$$
$$\underline{\underline{K}}_{22} = \int_\Omega e\left((\lambda+2\mu)\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_2}\otimes\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_2} + \mu\,\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_1}\otimes\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_1}\right)\mathrm{d}x_1\mathrm{d}x_2\,,$$
$$\underline{\underline{K}}_{12} = \int_\Omega e\left(\lambda\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_1}\otimes\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_2} + \mu\,\frac{\mathrm{d}\underline{\Phi}_1(\underline{x})}{\mathrm{d}x_2}\otimes\frac{\mathrm{d}\underline{\Phi}_2(\underline{x})}{\mathrm{d}x_1}\right)\mathrm{d}x_1\mathrm{d}x_2\,,$$
$$\underline{\underline{M}}_{ii} = \int_\Omega e\rho\,\underline{\Phi}_i(\underline{x})\otimes\underline{\Phi}_i(\underline{x})\,\mathrm{d}x_1\mathrm{d}x_2 \quad (i=1,2)\,,$$

and operators in time are given by

$$\begin{aligned}\underline{\underline{A}} &= \sum_{i=1}^{N_{slab}}\int_{I_i}\underline{\Psi}(t)\otimes\frac{\mathrm{d}\underline{\Psi}(t)}{\mathrm{d}t}\,\mathrm{dt} + \underline{\Psi}(t_0^+)\otimes\underline{\Psi}(t_0^+)\,,\\
&+\sum_{i=2}^{N_{slab}}\left(\underline{\Psi}(t_{i-1}^+)\otimes\underline{\Psi}(t_{i-1}^+) - \underline{\Psi}(t_{i-1}^+)\otimes\underline{\Psi}(t_{i-1}^-)\right),\\
\underline{\underline{B}} &= \sum_{i=1}^{N_{slab}}\int_{I_i}\underline{\Psi}(t)\otimes\underline{\Psi}(t)\,\mathrm{dt}\,.\end{aligned}$$

See Appendix A for details on the definition of the right hand side RHS.

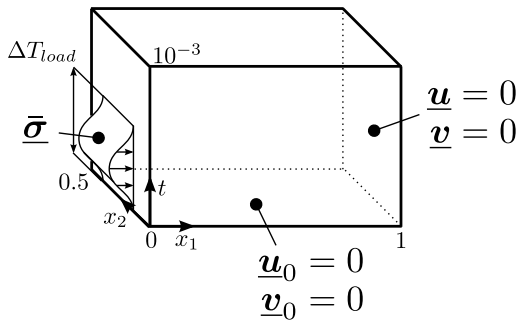

Figure 2: Test case description.

6. Numerical results

In this section, we investigate ability of space-time separated representation to approximate the transient solution of an elastodynamic problem, with decompositions of low rank. We consider transient responses due to impact loads whose duration varies from 0.01 to 1 ms. Comparisons between *a posteriori* and *a priori* low-rank approximations are made, and convergence of the new (IMR)PGD algorithm is analyzed.

6.1. Test-case description

The test-case describes the propagation of elastic waves in a rectangular medium submitted to impact loads.

6.1.1. Geometry, material and boundary conditions

Spatial domain is fixed to $\Omega_1 \times \Omega_2 = [0, 1]\text{m} \times [0, 0.5]\text{m}$ and thickness is $e = 0.1\text{m}$. Lamé's constants λ and μ are given in function of Young modulus E and Poisson's coefficient ν with $E = 2 \times 10^{11}\text{N/m}^2$ and $\nu = 0.3$. Density is taken as $\rho = 8000$ kg/m^3. Simulations are performed until $T = 10^{-3}s$. Also, null displacement and velocity are imposed on the right side of the medium and zero initial conditions are taken (see Figure 2). An impact load is applied on the middle left side of the medium. Its expression is given by

$$\underline{p}(x_1, x_2, t) = \begin{cases} 10^8(1 + \sin(\frac{2\pi}{\Delta T_{load}} t - \frac{\pi}{2}))\underline{e}_1 \text{ N/m}^2 & \forall t < \Delta T_{load} \\ \underline{0} & \forall t \geq \Delta T_{load} \end{cases}$$

$$\forall (x_1, x_2) \in \{0\} \times [0.125, 0.375]\,.$$

We vary the time ΔT_{load} of application of the load from 0.01 to 1 ms (we choose $\Delta T_{load} = 0.01 - 0.05 - 0.2 - 1$ ms).

6.1.2. Approximation spaces

Displacement and velocity fields are approximated with linear finite elements (continuous in space and piecewise continuous in time). This introduces a first source of error compared to the exact solution. Here, we do not consider this error and fix it to the same value for all test cases. The discretization error associated with the solutions $\{\mathbf{u}\} = (\mathbf{u}_1, \mathbf{u}_2)'$ and $\{\mathbf{v}\} = (\mathbf{v}_1, \mathbf{v}_2)'$ on a given mesh is calculated thanks to the following error indicator :

$$err_{DISC}(\{\mathbf{u}\}) = \frac{\left\| \{\mathbf{u}_e\} - \{\mathbf{u}\} \right\|_2}{\left\| \{\mathbf{u}_e\} \right\|_2}\,, \tag{42}$$

where $\{\mathbf{u}_e\} = (\mathbf{u}_{1e}, \mathbf{u}_{2e})'$ contains the values of the exact solution calculated on a finer mesh. We choose discretization parameters such that the discretization error related to the displacement field is less than 4% for all test cases (see Table 1). Notice that fixing the discretization error related to the displacement field to a certain value does not necessary ensure that the discretization error related to the velocity field is also smaller than this value. One should use different approximation spaces for both fields in order to be able to choose a particular error level for each field. This may be possible within the present strategy but we do not consider it here.

6.1.3. Space-time solutions

First, we compute space-time solutions with an incremental procedure in time. The solution obtained for the test case $\Delta T_{load} = 0.01$ ms is displayed at given time instants on Figure 3. It shows propagation of elastic waves in the medium due to the impact load. Displacement and velocity solutions are shown in space-time domain on Figure 4 for all test cases. One can notice that the transient solution is more and more localized over the space-time domain as the duration of the impact decreases.

6.2. A posteriori low-rank approximation

Since space-time solutions $\{\mathbf{u}\}$ and $\{\mathbf{v}\}$ have been calculated in a previous step, we can compute optimal low-rank approximations of these solutions *a posteriori*. These low-rank approximations are the POD of the full space-time solutions. They are calculated thanks to classical Singular Value Decomposition algorithm (SVD).

6.2.1. Errors due to low-rank approximation

Low-rank approximations of the displacement and velocity fields are calculated following definition (23). Then, we evaluate the decomposition error due to the rank-M approximation $\{\mathbf{u}\}_M$ (resp. $\{\mathbf{v}\}_M$) of the space space-time solution $\{\mathbf{u}\}$ (resp. $\{\mathbf{v}\}$) with the following indicator :

$$err(\{\mathbf{u}\}_M) = \frac{\left\| \{\mathbf{u}\} - \{\mathbf{u}\}_M \right\|_2}{\left\| \{\mathbf{u}\} \right\|_2}\,. \tag{43}$$

Decomposition errors obtained for all test cases are depicted in Figure 5 in function of the decomposition rank M. One can notice that higher decomposition ranks are needed in order to get a given decomposition error, when the duration of the impact decreases.

Remark 8. *The length of the time interval (denoted T), and consequently the number of waves reflections on boundary, has not a strong influence on the approximability using low rank approximations (detailed results can be found in [11]). This means that low rank approximations will provide a better memory gain if we increase the length of the time interval.*

6.2.2. Optimal memory gain

Memory gain related to storage requirements of a full space-time solution compared to its low-rank approximation, strongly depends on the size of the approximation space and the rank.

ΔT_{load} (ms)	Δt (ms)	$N(S) \times N(T)$	$err_{DISC}(\{\mathbf{u}\})$ (%)	$err_{DISC}(\{\mathbf{v}\})$ (%)
1	0.025	40×80	< 2	< 2
0.2	0.00625	544×320	< 2	< 5
0.05	0.0015625	8320×1280	< 2	< 9
0.01	0.000625	51520×3200	< 4	< 25

Table 1: Discretization parameters and associated discretization errors for all test cases. Parameters in space are fixed such that $\Delta x_1 = \Delta x_2 = c_L \Delta t$ where $c_L = \sqrt{E/\rho} = 5000$ m/s is the velocity of longitudinal waves.

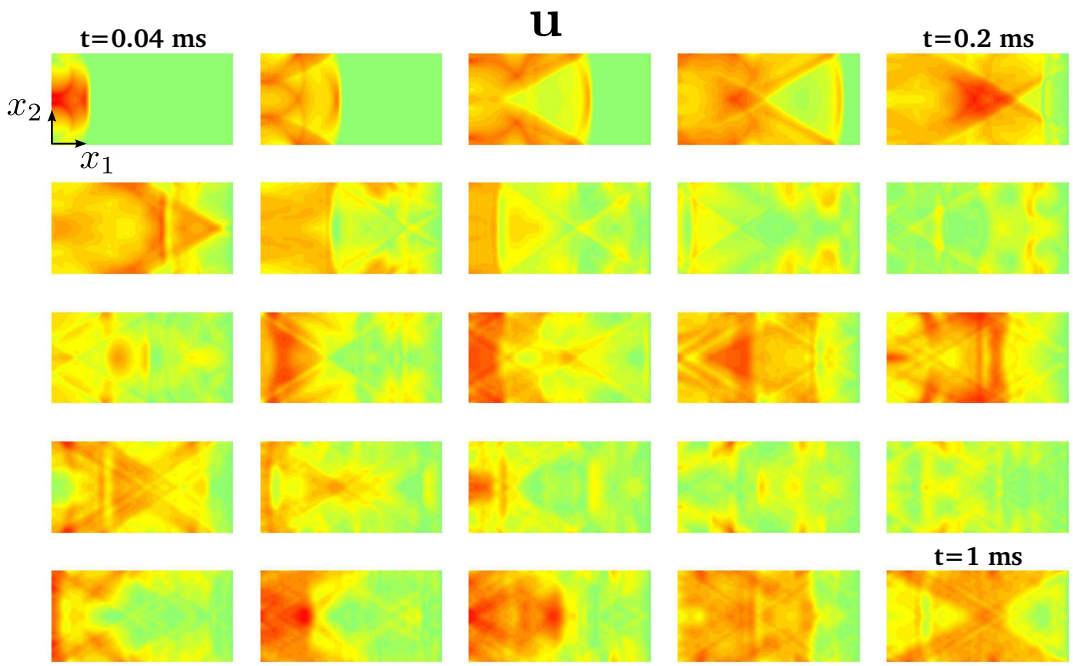

Figure 3: Amplitude of the displacement field at given time instants for the test case $\Delta T_{load} = 0.01$ ms.

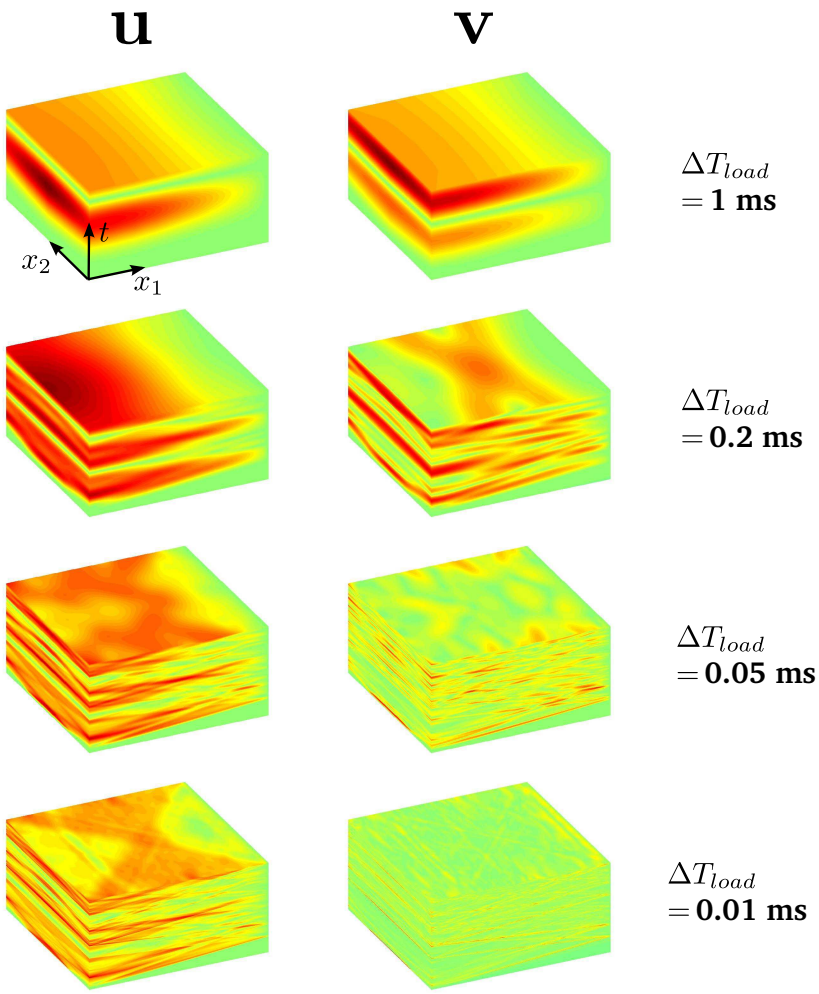

Figure 4: Space-time solutions for all test cases. Amplitude of displacement and velocity fields are shown.

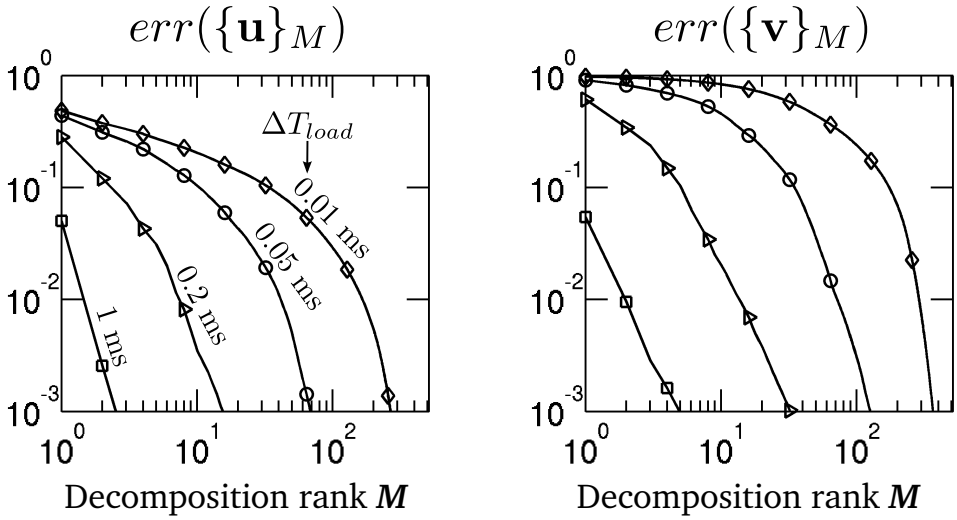

Figure 5: Decomposition error associated with the optimal low-rank approximation calculated *a posteriori* (POD) in function of the decomposition rank M. Errors obtained for all test cases and both displacement and velocity fields are shown for different ΔT_{load}. Discretization parameters are given in Table 1.

We numerically found that the error between the discrete space-time solution and its best approximation of rank M does not depend on the size of the mesh, provided that the mesh is fine enough [11]. For a given rank M, it is not necessary to choose a too large approximation space. Doing so will not decrease the error between the exact (continuous) solution and its (discrete) rank-M approximation. Therefore, we introduce in this section an "optimal memory gain" that takes into account both discretization and decomposition errors.

We propose to evaluate the memory gain related to the storage requirements of a full space-time solution on a given mesh and its low-rank approximation of same accuracy. To this end, we are looking for the minimal decomposition rank M for which decomposition error is smaller than discretization error (given in Table 1). Then, we calculate the memory gain as the ratio between the memory required to store the full solution and the memory required to store its low-rank approximation of same accuracy. That is, the optimal memory gain is defined as :

$$\text{memory gain} = \frac{N(S) \times N(T)}{M \times (N(S) + N(T))} \tag{44}$$

where M is the minimal integer such that $err(\{\mathbf{u}\}_M) \leq err_{DISC}(\{\mathbf{u}\})$.

Memory gains obtained with the displacement field for all test cases are summarized in Table 2. It shows that even if a higher rank is needed (for decomposition error to be lower than discretization error), memory gain gets better when duration of impact decreases. This is due to the finer space-time mesh needed at fixed discretization error in case of smaller duration of impact load.

ΔT_{load}	M	**mem. gain**
1	2	**13**
0.2	6	**34**
0.05	32	**35**
0.01	81	**37**

Table 2: Optimal memory gain factor associated with the space-time separated representation of the displacement field. The rank M is the minimal integer such that decomposition error $err(\{\mathbf{u}\}_M)$ is smaller than discretization error $err_{DISC}(\{\mathbf{u}\})$.

Thus, for cases we treated, *a posteriori* space-time low-rank representation allows decreasing memory required to store solutions of transient elastodynamic models from more than one order of magnitude[4]. Question is now : can we calculate good approximations of these optimal decompositions without calculating and storing the full space-time solution in a preliminary step ?

6.3. A priori low-rank approximations

We now construct low-rank approximations of the displacement and velocity fields *a priori*, that is without knowing $\{\mathbf{u}\}$ and $\{\mathbf{v}\}$. We compare accuracy of classical and ideal minimal residual-based PGDs with the optimal low-rank approximation computed *a posteriori* (that is the POD). The classical PGD is computed with a direct algorithm which is an extension to multi-field models of the algorithm called "Subspace iterations algorithm" in [36]. It is denoted by (R)PGD-S and allows finding the global minimizer defined in problem (26). For the computation of the ideal approach, pseudo-code is given in Algorithm 1. Convergence of this algorithm is studied in details in section 6.4 and we denote it by (IMR)PGD. It gives an approximation of the decomposition defined in problem (27) whose accuracy depends on the rank R of the approximation of the auxiliary problem's solution.

6.3.1. Influence of discretization parameters

First, we compare the accuracy of the obtained low-rank approximations at a fixed rank $M = 10$ for the test case $\Delta T_{load} = 0.2$ ms. In order to exhibit the influence of the space-time problem condition number (see section 2.3) on the decomposition accuracy, calculations are made for different discretization parameters. Results are summarized in Table 3. One can notice that (R)PGD-S algorithm converges to a poorly accurate low-rank approximation compared to the optimal decomposition given by the POD. Moreover, the accuracy of the decomposition calculated with (R)PGD-S algorithm deteriorates as the size of the discretization space increases. On the contrary, the (IMR)PGD algorithm allows to find the optimal low-rank approximation with a very good accuracy. And accuracy of the low-rank approximation obtained with (IMR)PGD is very less affected by the size of the discretization space. Indeed, as explained in [8], the choice made for the ideal norm (used to minimize the residual) implies that the space-time problem is ideally conditioned.

6.3.2. Influence of impact loads duration

Then, we compare the accuracy of the low-rank approximations given by (IMR)PGD algorithm in function of the auxiliary rank R as the duration of the impact load decreases (we use $\Delta T_{load} = 0.05 - 0.2 - 1$ ms and discretization parameters given in Table 1). The decomposition is calculated with the (IMR)PGD algorithm for different values of decomposition rank M (we take $M = 1, 2, 4, 8, 16, 32, 64$) and different values of auxiliary rank R (we take $R = 1, 2, 4, 8, 16$). As can be noticed on Figure 6, (IMR)PGD algorithm allows finding a very good approximation of the optimal decomposition (given by POD) with a very small number of auxiliary modes, even for high decomposition rank M and small duration of impact (see the points for $M = 64$ and $\Delta T_{load} = 0.05$ ms). This means that the auxiliary problem's solution can be well approximated with a very low-rank approximation, even for small duration of the impact load. However, as it will be illustrated in the next section, the convergence of (IMR)PGD algorithm strongly depends on the auxiliary rank R and more iterations are needed for the algorithm to converge when it is used with small values of R.

Remark 9. *We also compared accuracy of classical Reduced Order Models - ROMs, namely truncated modal basis and POD-Snapshot approaches, with the best rank-M approximation of*

[4]Better gains (more than two orders of magnitude) were obtained by separating spatial variables in addition to the time variable but these results are not presented in the present paper where we only consider low-rank approximations of second-order tensors.

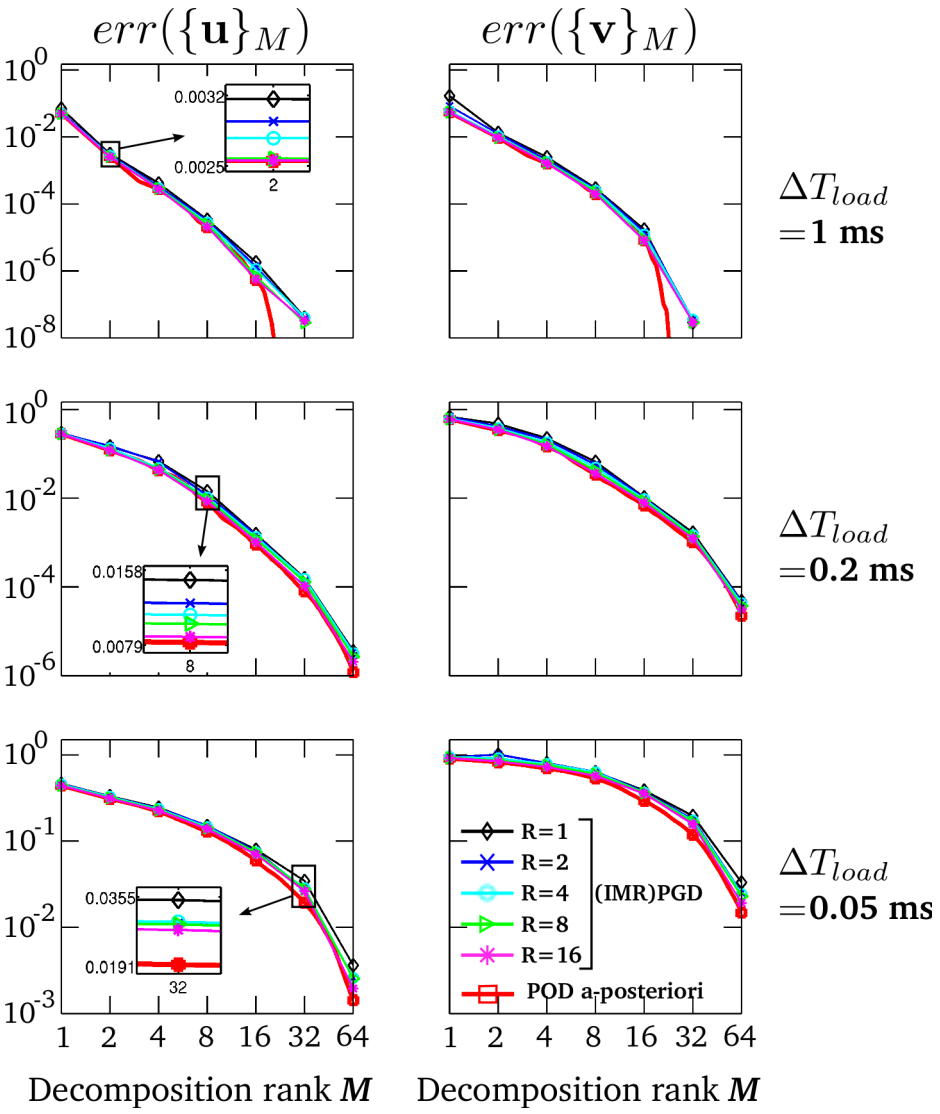

Figure 6: Influence of auxiliary rank R on the accuracy of the decompositions obtained *a priori* with (IMR)PGD in function of the decomposition rank M, compared to decomposition errors obtained *a posteriori* with POD.

	$err(\{\mathbf{u}\}_{10})$					
$N(S) \times N(T)$	POD	(R)PGD-S	(IMR)PGD			
			R=5	R=10	R=20	R=40
144×160	2.5e-03	3.5e-02	3.4e-03	2.9e-03	2.6e-03	2.5e-03
544×320	3.5e-03	3.9e-02	4.8e-03	3.9e-03	3.6e-03	3.5e-03
2112×640	3.6e-03	9.8e-02	4.4e-03	4.0e-03	3.8e-03	3.7e-03
8320×1280	3.6e-03	1.0e+00	4.4e-03	4.0e-03	3.7e-03	3.6e-03
	$err(\{\mathbf{v}\}_{10})$					
$N(S) \times N(T)$	POD	(R)PGD-S	(IMR)PGD			
			R=5	R=10	R=20	R=40
144×160	1.4e-02	1.3e-01	1.8e-02	1.6e-02	1.5e-02	1.4e-02
544×320	2.0e-02	1.4e-01	2.7e-02	2.4e-02	2.1e-02	2.1e-02
2112×640	2.2e-02	3.1e-01	3.0e-02	2.6e-02	2.3e-02	2.2e-02
8320×1280	2.1e-02	9.7e-01	4.5e-02	3.2e-02	2.4e-02	2.2e-02

Table 3: Comparison of decomposition errors obtained *a posteriori* (with POD) and *a priori* (with (R)PGD-S and (IMR)PGD for different auxiliary ranks R) in function of discretization parameters. Errors are calculated for a fixed decomposition rank $M = 10$ and the test case $\Delta T_{load} = 0.2$ ms.

the full space-time solution, in case of transient dynamic problems. Detailed results can be found in [11]. We numerically showed that approximations obtained with classical ROMs are less accurate than the best rank-M approximation defined in the present paper. For transient dynamic cases we treated, these approaches require to calculate and store almost all structural modes (for the truncated modal basis approach) or snapshots at nearly all time steps of the reference temporal mesh (for POD-Snapshot approach) in order to get accurate approximations. On the other hand, as shown in section 6.3.1, classical PGD algorithms also fail to give an accurate rank-M approximation of transient dynamic solutions since space-time operators of such problems are ill-conditioned. For a given decomposition error, classical minimal residual PGD requires to compute far more space-time modes than necessary compared to the best rank-M approximation. Then, the main interest of the (IMR)PGD algorithm is to allow a priori calculation of a good approximation of the best rank-M approximation without having to calculate and store much more space-time modes than necessary.

6.4. Convergence of (IMR)PGD algorithm

In this section, we numerically study the convergence of (IMR)PGD algorithm. We first describe the algorithm and its parameters. Then, their influence on the convergence of the algorithm is analyzed. A stopping criterion is finally proposed and analyzed. Notice that, here, $\{\mathbf{u}\}_M$ is used to denote the rank-M approximation of a tuple whose components are the rank-M approximations of the components of the displacement and velocity fields, that is $\{\mathbf{u}\}_M = (\mathbf{u}_{1M}, \mathbf{u}_{2M}, \mathbf{v}_{1M}, \mathbf{v}_{2M})'$.

6.4.1. Algorithm parameters

Pseudo-code for (IMR)PGD is given in Algorithm 1. The algorithm depends on the following steps :

- Step (4) is used to give a rank-R approximation of the auxiliary problem's solution. It is computed with the multi-field version of the progressive Galerkin-based PGD algorithm described in [10].
- Step (5) gives an optimal rank-M approximation of a tensor given in low-rank format (that is the tensor $\{\mathbf{u}\}_M + [\mathbf{A}]' \overset{\mathrm{D}}{\cdot} \{\mathbf{y}\}_R$). Since this tensor is already given in low-rank format, computational cost of this step is negligible compared to the cost of the auxiliary step (4).

These two steps are related to minimization problems. Algorithms used to solve these problems are stopped once a criterion based on stagnation of the minimized functional is reached. The value of the stopping criterion has not a decisive influence on the accuracy of the decomposition $\{\mathbf{u}\}_M$ obtained at the end of Algorithm 1. A small number of iterations (typically less than ten) can be used for the progressive computation of each mode, see Appendix B for more details. In the following, we fixed the value of these stopping criterions so that the numerical error associated with the construction of the low-rank approximations at step (4) and (5) of Algorithm 1 do not influence accuracy of $\{\mathbf{u}\}_M$.

Algorithm 1 $\{\mathbf{u}\}_M$ = **(IMR)PGD**$([\mathbf{A}], \{\mathbf{b}\}, M, R)$ – Direct construction

1: $\{\mathbf{u}\}_M = \mathbf{0}$
2: **for** $k = 1$ to k_{max} **do**
3: $\quad \{\mathbf{r}\} = \{\mathbf{b}\} - [\mathbf{A}] \overset{\mathrm{D}}{\cdot} \{\mathbf{u}\}_M$
4: $\quad \{\mathbf{y}\}_R$ = **(G)PGD-MF**$([\mathbf{A}] \overset{\mathrm{D}}{\cdot} [\mathbf{A}]', \{\mathbf{r}\}, R)$
5: $\quad \{\mathbf{u}\}_M$ = **SVD**$(\{\mathbf{u}\}_M + [\mathbf{A}]' \overset{\mathrm{D}}{\cdot} \{\mathbf{y}\}_R, M)$
6: $\quad$ Check convergence of $\{\mathbf{u}\}_M$
7: **end for**

Key parameters of Algorithm 1 as regards the convergence of the algorithm are the auxiliary rank R and the decomposition rank M. To study the influence of these parameters on the convergence of the algorithm, we compute the multi-field error given by $\left\| \{\mathbf{u}\} - \{\mathbf{u}\}_M^{(k)} \right\|_2$ at all iterations k. We recall that this is the error we try to minimize for a given rank M. Also, we compare this error with the value of $\left\| \{\mathbf{y}\}_R^{(k)} \right\|_{[\mathbf{A}] \overset{\mathrm{D}}{\cdot} [\mathbf{A}]'}$ obtained at all iterations k. As explained in subsection 3.4,

the value of $\left\| \{\mathbf{y}\}_R^{(k)} \right\|_{[\mathbf{A}]^{D.}[\mathbf{A}]'}$ gives a good estimate of the error $\left\| \{\mathbf{u}\} - \{\mathbf{u}\}_M^{(k)} \right\|_2$, provided the auxiliary rank R is high enough. For all calculations we made (see Figures 7 to 9), we observe the same behavior of the algorithm: in a first stage, the error decreases in a linear way as iterations increase, and then it stagnates around a particular value.

6.4.2. Influence of decomposition rank M

As can be noticed on Figure 7, the order of convergence of the error does not depend on the decomposition rank M. Only the value to which the error stagnates depends on M. This can be easily understood by the fact that the error $\left\| \{\mathbf{u}\} - \{\mathbf{u}\}_M \right\|_2$ is smaller for higher values of decomposition rank M. Then, the number of iterations needed to reach stagnation of the error is higher when computing decomposition of higher rank M.

6.4.3. Influence of auxiliary rank R

On the contrary, the order of convergence of the error depends strongly on the auxiliary rank R. As can be observed on Figure 8, higher the auxiliary rank R is, the faster the algorithm converges. Indeed, using a higher number of auxiliary modes yields a more accurate approximation of the auxiliary solution, and then to a smaller number of algorithm iterations. Also, we notice that the error reaches its stagnation value in very few iterations provided the auxiliary rank R is high enough. As an example, less than twenty iterations are necessary to reach stagnation of the error for the test case $1/\Delta T_{load} = 20$ kHz using $R = 16$ auxiliary modes for computing the decomposition of rank $M = 64$.

6.4.4. Influence of discretization parameters

The convergence of the algorithm does not depend much on the discretization parameters. The same order of convergence and the same number of iterations necessary to reach stagnation of the error are observed whatever the discretization parameters are (see the convergence curves depicted in Figure 9).

6.4.5. Stopping criterion

As can be noticed on Figures 7 to 9, the value of $\left\| \{\mathbf{y}\}_R^{(k)} \right\|_{[\mathbf{A}]^{D.}[\mathbf{A}]'}$ reflects the behavior of the true error $\left\| \{\mathbf{u}\} - \{\mathbf{u}\}_M^{(k)} \right\|_2$ as k increases, provided the auxiliary rank R is high enough. Then, to stop algorithm iterations, we use a heuristic criterion that allows determining a number of iterations after which we can consider there is stagnation of $\left\| \{\mathbf{y}\}_R^{(k)} \right\|_{[\mathbf{A}]^{D.}[\mathbf{A}]'}$. However, we may fail to detect stagnation of the error if a too small auxiliary rank R is used. Indeed, high oscillations of $\left\| \{\mathbf{y}\}_R^{(k)} \right\|_{[\mathbf{A}]^{D.}[\mathbf{A}]'}$ around its stagnation value are observed if the auxiliary rank R is too small. These oscillations may be attenuated by choosing adaptively the auxiliary rank R during iterations, as done in [8].

Remark 10. *Our PGD computational technique is still at a early stage of development and our implementation should be optimized. Therefore, we did not compared calculation times of our PGD algorithm with calculation times of the incremental solver. Nevertheless, we can compare complexity of both approaches. To this end, we denote* $\boldsymbol{lin}(n)$ *the complexity*[5] *associated with solution of a square linear system of size n. To simplify, consider a single field problem discretized with* n_S *degrees of freedom and* n_T *time steps. Then, the order of magnitude of the complexity associated with an incremental solution is* $n_T \times \boldsymbol{lin}(n_S)$ *(if we use an implicit time scheme as is the case in the present paper). The most expensive step of the (IMR)PGD algorithm is the computation of the rank-R approximation of the auxiliary problem's solution (using a progressive Galerkin PGD algorithm). This cost is due to* k_{max}^{aux} *solutions of two linear systems (one in space and an other in time). Neglecting the cost of these linear systems assembly, the complexity of one auxiliary step is thus of the order of magnitude of* $R \times k_{max}^{aux} \times (\boldsymbol{lin}(n_S) + \boldsymbol{lin}(n_T))$*. This auxiliary step being computed* k_{max} *times, we obtain* $k_{max} \times R \times k_{max}^{aux} \times (\boldsymbol{lin}(n_S) + \boldsymbol{lin}(n_T))$ *for the order of magnitude of the complexity of the (IMR)PGD algorithm. More details can be found in [11]. Assume now that* $n_S = n_T = n$*, we see that the (IMR)PGD algorithm will be faster than the incremental solver only if* $k_{max} \times R \times k_{max}^{aux} << n/2$*. This condition is not satisfied for all numerical experiments we did and the complexity of the (IMR)PGD algorithm is only of the order of magnitude of (or bigger than) the complexity of the incremental solver. In fact, the potential of the (IMR)PGD algorithm, in terms of computational time reduction, will be revealed if we increase the problem dimensionality (by splitting spatial variables in addition to the time variable [9] or in uncertainty quantification context [35]) or if multiple transient dynamic simulations have to be performed (as is the case for non-linear problems [30] or in optimization context [31]).*

7. Conclusions and Perspectives

In this paper, we have illustrated the ability of space-time separated representations to approximate the solution of elastodynamic models with two dimensions in space. We show that transient responses due to impact loads can be well approximated with decompositions of low rank.

The optimal low-rank approximation has first been defined *a posteriori* with respect to a target norm. Using a classical canonical norm, this optimal decomposition is simply the POD of the full space-time solution. The aim of this paper was to compute this optimal approximation *a priori* that is without knowing the full space-time solution. To this end, we have extended to multi-field models an algorithm (called "Ideal Minimal Residual PGD") introduced in [8] that allows finding *a priori* a very good approximation of the best low-rank approximation with respect to the target norm.

The accuracy of the obtained decomposition depends on the low-rank approximation of an auxiliary problem's solution. Numerical results have demonstrated the convergence of the algorithm even for very small values of the auxiliary rank, showing that the auxiliary problem's solution can be well approxi-

[5] Notice that PGD algorithms preserve properties of spatial and temporal operators such as sparsity. Therefore, we can use the same linear solver complexity (denoted by $\mathbf{lin}(n)$) to compare both approaches.

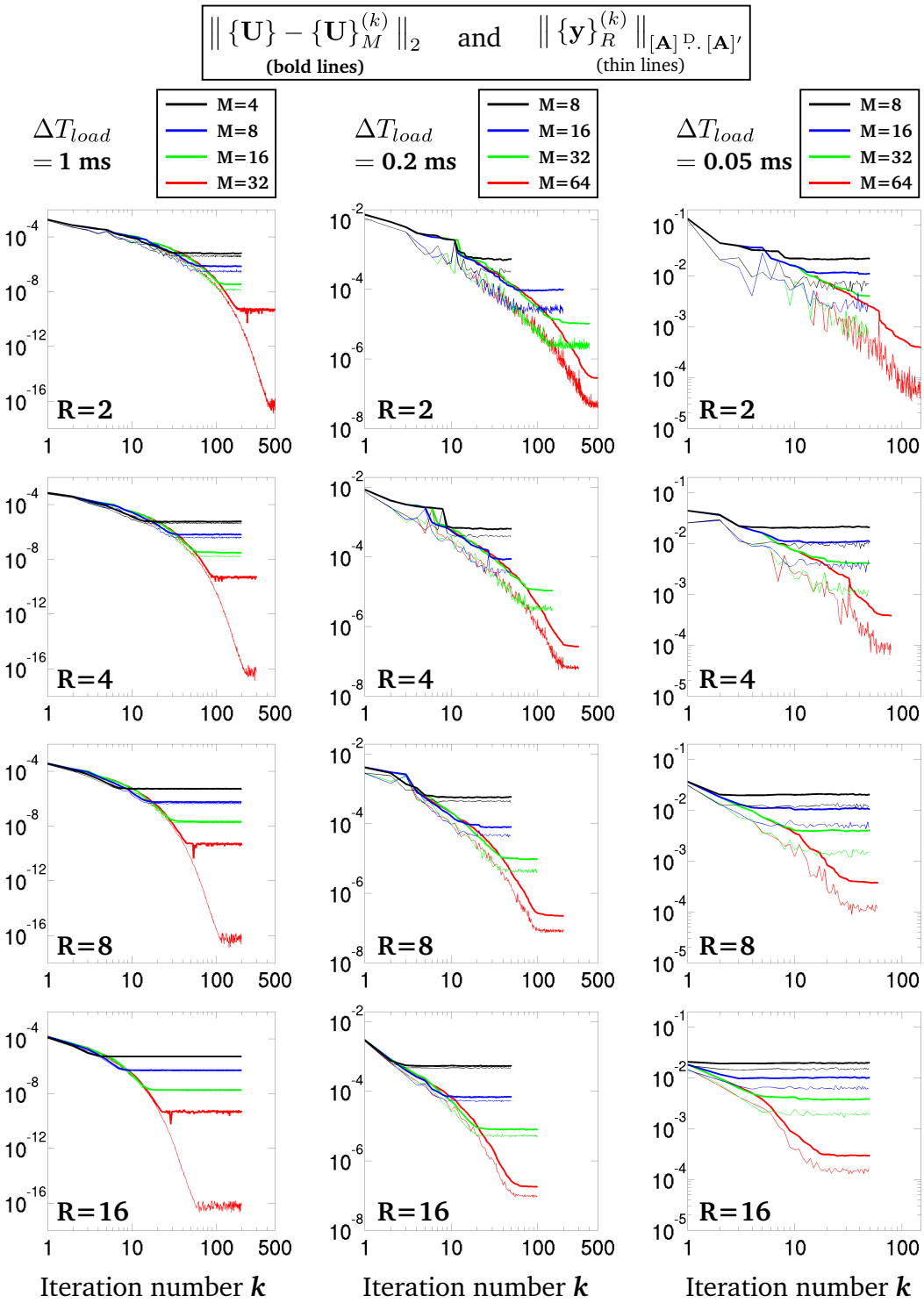

Figure 7: Influence of decomposition rank M on the convergence of the multi-field (IMR)PGD algorithm.

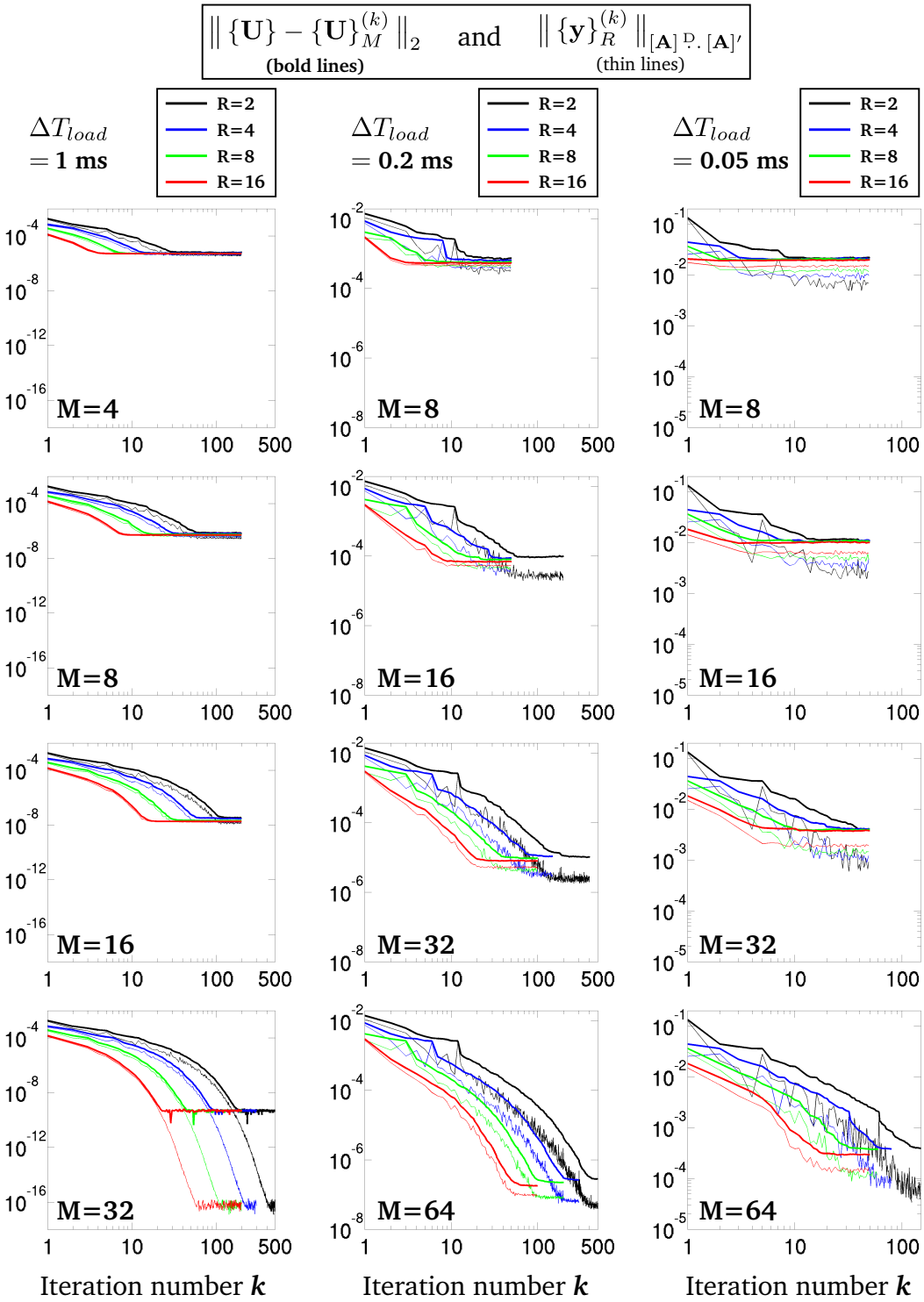

Figure 8: Influence of auxiliary rank R on the convergence of the multi-field (IMR)PGD algorithm.

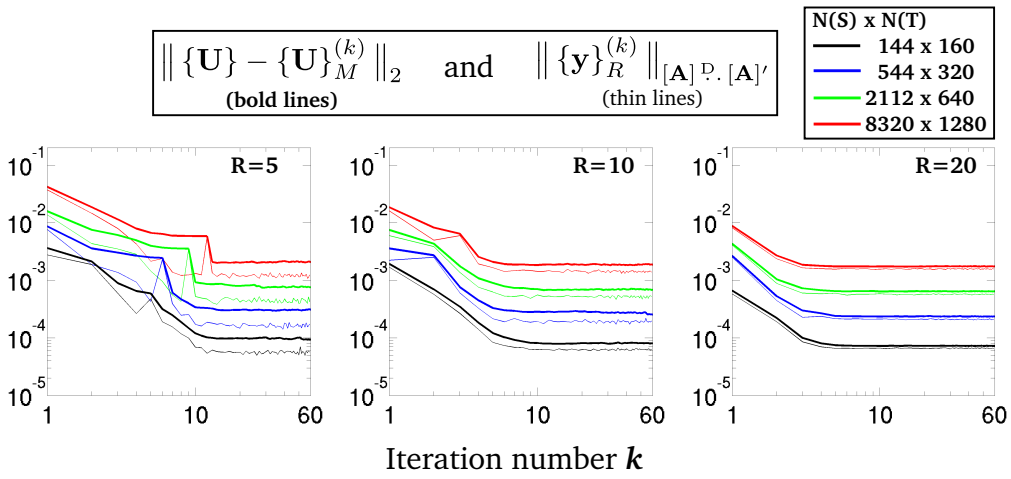

Figure 9: Influence of discretization parameters on the convergence of the multi-field (IMR)PGD algorithm. Errors are calculated for a fixed decomposition rank $M = 10$ and the test case $\Delta T_{load} = 0.2$ ms.

mated with a very small number of modes. However, convergence of the algorithm strongly depends on this auxiliary rank and more auxiliary modes should be used to decrease the number of iterations necessary for the algorithm to converge. Also, the computed approximation of the auxiliary solution can be used to provide an accurate lower bound of the low-rank approximation error with respect to the target norm, assuming the auxiliary rank is high enough.

Current developments focus on finding an efficient procedure to choose the auxiliary rank adaptively during the iterative process, so that strict lower and upper bounds of the decomposition error can be calculated. Also, acceleration strategies based on multi-grid methods and exploiting convergence property of the algorithm are under investigations [38, 13].

Acknowledgment

This work was supported by the French National Research Agency under grant ANR-10-COSI-0006.

Appendix A. Right hand side derivation

In this section, details about the definition of the right hand side of equation (41) are given. This right hand side is related to Dirichlet conditions $\underline{\bar{u}}$ and $\underline{\bar{v}}$, to Neumann condition $\underline{p}$ and to initial conditions $\underline{u}_0$ and $\underline{v}_0$. It can be written in general form as

$$RHS = \begin{bmatrix} \mathbf{b}_1^{u_0} \\ \mathbf{b}_2^{u_0} \\ \mathbf{b}_1^{v_0} + \mathbf{b}_1^{p} - \mathbf{b}_1^{\bar{u}} - \mathbf{b}_1^{\bar{v}} \\ \mathbf{b}_2^{v_0} + \mathbf{b}_2^{p} - \mathbf{b}_2^{\bar{u}} - \mathbf{b}_2^{\bar{v}} \end{bmatrix} . \tag{A.1}$$

In order to identify each contribution to the right hand side, all boundary conditions must be given in a low-rank format. If this is not the case, truncated SVD could be first applied to the data.

Each component $b_i(\underline{x}, t)$ of a given boundary field $\underline{b}(\underline{x}, t)$ (except initial conditions) are written under the following form : for $i = 1$ or 2,

$$b_i(\underline{x}, t) = \sum_{m=1}^{M_b(i)} b_{im}^S(\underline{x})\, b_{im}^T(t) \quad \text{with} \quad \underline{x} \in \partial\Omega_u \text{ or } \partial\Omega_\sigma,\ t \in I\ .$$

Then, the boundary fields are discretized as follows :

$$b_i(\underline{x}, t) = (\underline{\tilde{\Phi}}(\underline{x}) \otimes \underline{\Psi}(t)) \,\underline{\underline{\mathrm{D}}}\, \mathbf{b}_i \quad \text{with} \quad \mathbf{b}_i = \sum_{m=1}^{M_b(i)} \underline{b}_{im}^S \otimes \underline{b}_{im}^T\ ,$$

and introduced in the weak formulation (34). This allows identifying all contributions $\mathbf{b}_1^{\bar{\sigma}}$, $\mathbf{b}_2^{\bar{\sigma}}$, $\mathbf{b}_1^{\bar{u}}$, $\mathbf{b}_2^{\bar{u}}$, $\mathbf{b}_1^{\bar{v}}$, $\mathbf{b}_2^{\bar{v}}$ due to external loads, imposed displacement and velocity. In order to identify contributions due to initial displacement and velocity $\mathbf{b}_1^{u_0}$, $\mathbf{b}_2^{u_0}$, $\mathbf{b}_1^{v_0}$, $\mathbf{b}_2^{v_0}$, the following approximation of initial fields must be introduced in the weak formulation (34) :

$$\forall i = 1 \ldots 2, \quad b_{i0}(\underline{x}) = \underline{\Phi}(\underline{x})\,.\,\underline{b}_{i0} \quad \text{with} \quad \underline{x} \in \Omega$$

Appendix B. Stopping criterion of the auxiliary PGD

In this section, details about the stopping criterion used to stop the algorithm that computes the auxiliary low-rank approximation at step (4) of Algorithm 1 are given. This step is computed with the algorithm $\{\mathbf{y}\}_R$ = **(G)PGD-MF**($[\mathbf{A}] \,\underline{\underline{\mathrm{D}}}\, [\mathbf{A}]', \{\mathbf{r}\}, R$) that gives a rank-$R$ approximation of the solution of the linear system with operator $[\mathbf{A}] \,\underline{\underline{\mathrm{D}}}\, [\mathbf{A}]'$ and right hand side $\{\mathbf{r}\}$. This algorithm is related to the minimization of the following functional (see [10]):

$$J_{aux} : \begin{cases} \mathsf{S}^F \to \mathbb{R} \\ \{\mathbf{y}\} \mapsto \frac{1}{2}\{\mathbf{y}\} \,\underline{\underline{\mathrm{D}}}\, [\mathbf{A}] \,\underline{\underline{\mathrm{D}}}\, [\mathbf{A}]' \,\underline{\underline{\mathrm{D}}}\, \{\mathbf{y}\} - \{\mathbf{y}\} \,\underline{\underline{\mathrm{D}}}\, \{\mathbf{r}\} \end{cases} \tag{B.1}$$

The decomposition is constructed in a progressive way (see Remark 2). For a given decomposition $\{\mathbf{y}\}_{R-1}$ of rank $R-1$, the new rank-one enrichment $\{\mathbf{w}\} \in \mathsf{S}_1^F$ is calculated such that $J(\{\mathbf{y}\}_{R-1} + \{\mathbf{w}\})$ is minimum and $\{\mathbf{y}\}_R = \{\mathbf{y}\}_{R-1} + \{\mathbf{w}\}$. This minimization problem is solved with an alternating minimization

algorithm and a criterion based on stagnation of the functional J_{aux} is used. At a given iteration (k), we calculate the following criterion :

$$\varepsilon_{aux}(k) = \frac{|J_{aux}(\{\mathbf{y}\}_{R-1} + \{\mathbf{w}\}^{(k)}) - J_{aux}(\{\mathbf{y}\}_{R-1} + \{\mathbf{w}\}^{(k-1)})|}{J_{aux}(\{\mathbf{y}\}_{R-1} + \{\mathbf{w}\}^{(k)})} \tag{B.2}$$

and we stop the iterative process once $\varepsilon_{aux}(k) < \varepsilon_{aux}^{max}$.

Influence of the stopping value ε_{aux}^{max} on the accuracy of decomposition $\{\mathbf{u}\}_M$ obtained for $M = 10$ and $R = 10$ at the end of Algorithm 1 is illustrated in Table B.4. It shows the decomposition error $err(\{\mathbf{u}\}_{10})$ in function of stopping values ε_{aux}^{max} and for different discretization parameters. Also, the mean number of alternating minimization iterations k_{aux}^{moy} necessary to reach the stopping criterion during the computation of each auxiliary mode is shown in parentheses in Table B.4. As can be noticed, only a small number of iterations (less than ten) for the computation of each auxiliary mode is sufficient to get the full converged decomposition $\{\mathbf{u}\}_{10}$ with a quite good approximation.

References

[1] Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory. *J. Non-Newtonian Fluid Mech.* **139** (2006) 153-176

[2] Ammar A, Mokdad B, Chinesta F, Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II: Transient simulation using space-time separated representations. *J. Non-Newtonian Fluid Mech.* **144** (2007) 98-121

[3] Ammar A. The Proper Generalized Decomposition: a powerful tool for model reduction. *Int. J. Mater. Form.* **3** (2010) 89-10

[4] Ammar A, Chinesta F, Falcó A. On the convergence of a Greedy rank-one update algorithm for a class of linear systems. *Arch. Comput. Methods Eng.* **17** (2010) 473-486

[5] Beylkin G, Mohlenkamp MJ. Algorithms for numerical analysis in high dimensions. *SIAM J. Sci. Comput.* **26** (2005) 2133-2159

[6] Beringhier M, Gueguen M, Grandidier JC. Solution of strongly coupled multiphysics problems using space-time separated representations - application to thermoviscoelasticity. *Arch. Comput Methods Eng* **17** (2010) 393-401

[7] Berkooz G, Holmes P, Lumley JL. The Proper Orthogonal Decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid Mech.* **25** (1993) 539-75

[8] Billaud-Friess M, Nouy A, Zahm O. A tensor approximation method based on ideal minimal residual formulations for the solution of high dimensional problems. arXiv:1304.6126

[9] Bognet B, Bordeu F, Chinesta F, Leygue A, Poitou A. Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. *Comput. Methods Appl. Mech. Engrg.* **201-204** (2012) 1-12

[10] Boucinha L, Gravouil A, Ammar A. Space-time Proper Generalized Decompositions for the resolution of transient elastodynamic models. *Comput. Methods Appl. Mech. Engrg.* **255** (2013) 67-88

[11] Boucinha L. Réduction de modèle a priori par séparation de variables espace-temps. Application en dynamique transitoire. *PhD thesis, INSA Lyon* (2013)

[12] Cances E, Ehrlacher V, Lelievre T. Greedy algorithms for high-dimensional non-symmetric linear problems. (2012) arXiv:1210.6688

[13] Cavin P, Gravouil A, Lubrecht A, Combescure A. Automatic energy conserving space-time refinement for linear dynamic structural problems. *Int. J. Numer. Meth. Engng.* **64** (2005) 304-321

[14] Cohen A, Dahmen W, Welper G. Adaptivity and variational stabilization for convection-diffusion equations. *ESAIM:Mathematical Modelling and Numerical Analysis* **46** (2012) 1247-1273

[15] Chinesta F, Ammar A, Lemarchand F, Beauchene P, Boust F. Alleviating mesh constraints: model reduction, parallel time integration and high resolution homogenization *Comput. Methods Appl. Mech. Engrg.* **197** (2008) 400-413

[16] Chinesta F, Ladevèze P, Cueto E. A short review on model order reduction based on Proper Generalized Decomposition. *Arch. Comput Methods Eng* **404** (2011) 395-404

[17] Dahmen W, Huang C, Schwab C, Welper G. Adaptive Petrov-Galerkin methods for first order transport equations. *Journal on Numerical Analysis* **50** (2012) 2420-2445

[18] Dumon A, Allery C, Ammar A. Proper General Decomposition (PGD) for the resolution of Navier-Stokes equations. *Journal of Computational Physics* **230** (2011) 1387-1407

[19] Dureisseix D, Ladevèze P, Schrefler BA. A computational strategy for multiphysics problems : application to poroelasticity. *Int J Numer Methods Eng* **56-10** (2003) 1489-1510

[20] Falcó A, Nouy A. A Proper Generalized Decomposition for the solution of elliptic problems in abstract form by using a functional Eckart-Young approach. *Journal of Mathematical Analysis and Applications* **376** (2011) 469-480

[21] Falcó A, Nouy A. Proper Generalized Decomposition for nonlinear convex problems in tensor Banach spaces. *Numer. Math.* (2012) **121** 503-530

[22] Falcó A, Hilario L, Montés N, Mora MC. Numerical strategies for the Galerkin Proper Generalized Decomposition method. *Mathematical and Computer Modelling* (2013) **57** 1694-1702

[23] Géradin M, Rixen D. Mechanical vibrations, theory and application to structural dynamics. *Wiley*, 1994

[24] Glösmann P, Kreuzer E. On the application of Karhunen-Loeve transform to transient dynamic systems. *Journal of Sound and Vibration* **328** (2009) 507-519

[25] Hackbusch W. Tensor Spaces and Numerical Tensor Calculus. *Springer*, 2012

[26] Hulbert GM, Hughes TJR. Space-time finite element methods for second-order hyperbolic equations. *Comput. Methods Appl. Mech. Eng.* **84** (1990) 327-348

[27] Hulbert GM. Computational structural dynamics. *Encyclopedia of Computational Mechanics, John Wiley & Sons*, 2004

[28] Kerschen G, Golinval JC, Vakakis AF, Bergman LA. The method of Proper Orthogonal Decomposition for dynamical characterization and order reduction of mechanical systems: An overview. *Nonlinear Dynamics* **41** (2005) 147-169

[29] Kunisch K, Volkwein S. Galerkin Proper Orthogonal Decomposition methods for parabolic problems. *Numerische Mathematik* **90** (2001) 117-148

[30] Ladevèze P. Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation. *Springer*, 1999

[31] Leygue A, Verron E. A first step towards the use of proper general decomposition method for structural optimization. *Arch. Comput Methods Eng* **17** (2010) 465-472

[32] Néron D, Dureisseix D. A computational strategy for thermo-poroelastic structures with a time-space interface coupling. *Int J Numer Methods Eng* **75-9** (2008) 1053-1084

[33] Néron D, Ladevèze P. Proper Generalized Decomposition for multiscale and multiphysics problems. *Arch. Comput Methods Eng* **17** (2010) 351-372

[34] Nouy A. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg.* **196** (2007) 4521-4537

[35] Nouy A. Proper Generalized Decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Arch. Comput Methods Eng* **17** (2010) 403-434

[36] Nouy A. A priori model reduction through Proper Generalized Decomposition for solving time-dependent partial differential equations. *Comput. Methods Appl. Mech. Engrg.* **199** (2010) 1603-1626

[37] Qu ZQ. Model Order Reduction Techniques with Application in Finite Element Analysis. *Springer-Verlag London* (2004)

[38] Sterck H, Miller K. An Adaptive Algebraic Multigrid Algorithm for Low-Rank Canonical Tensor Decomposition. *SIAM Journal on Scientific Computing* **35** (2013) 1-24

	$err(\{\mathbf{u}\}_{10})$ and k_{aux}^{moy} in parentheses			
$N(S) \times N(T)$	ε_{aux}^{max} =1e-01	ε_{aux}^{max} =1e-03	ε_{aux}^{max} =1e-05	ε_{aux}^{max} =1e-07
144×160	3.6e-03(7)	3.0e-03(24)	3.3e-03(44)	2.9e-03(58)
544×320	5.0e-03(7)	4.5e-03(25)	4.0e-03(42)	4.0e-03(58)
2112×640	4.8e-03(7)	4.3e-03(25)	4.0e-03(43)	4.0e-03(58)
8320×1280	1.3e-02(8)	4.1e-03(24)	4.0e-03(44)	4.0e-03(54)

Table B.4: Influence of stopping criterion ε_{aux}^{max} used at step (4) of Algorithm 1 on the accuracy of the obtained decomposition $\{\mathbf{u}\}_{10}$ for different values of discretization parameters. k_{aux}^{moy} is the mean number of alternating minimization iterations necessary to reach the stopping criterion during the computation of each auxiliary mode. Other parameters are maximum number of iterates $k_{max} = 10$, auxiliary rank $R = 10$ and test case $\Delta T_{load} = 0.2$ ms.