

RESEARCH ARTICLE

Open Access



# Proper Generalized Decomposition computational methods on a benchmark problem: introducing a new strategy based on Constitutive Relation Error minimization

Pierre-Eric Allier\*, Ludovic Chamoin and Pierre Ladevèze

\*Correspondence:  
allier@lmt.ens-cachan.fr  
LMT (ENS Cachan/CNRS/  
Université Paris Saclay), 61  
avenue du Président Wilson,  
94235 Cachan, France

## Abstract

First, the effectivity of classical Proper Generalized Decomposition (PGD) computational methods is analyzed on a one dimensional transient diffusion benchmark problem, with a moving load. Classical PGD methods refer to Galerkin, Petrov–Galerkin and Minimum Residual formulations. A new and promising PGD computational method based on the Constitutive Relation Error concept is then proposed and provides an improved, immediate and robust reduction error estimation. All those methods are compared to a reference Singular Value Decomposition reduced solution using the energy norm. Eventually, the variable separation assumption itself (here time and space) is analyzed with respect to the loading velocity.

**Keywords:** Model Reduction, Proper Generalized Decomposition (PGD), Separation of variables, Constitutive Relation Error (CRE)

## Notations

All continuous fields or spaces are in classical format  $u$ , while Finite Element approximations fields  $u_h$  benefit of an  $h$  index. Discretized terms, usually represented by vectors of nodal values, are bolded  $\mathbf{u}$  and matrices are in uppercase and double-struck  $\mathbb{U}$ .

## Background

Nowadays, numerical simulations constitute a common tool in science and engineering activities. It is especially used for prediction and decision making, or simply for a better understanding of physical phenomena. However, in order to give an accurate representation of the real world, a large set of parameters and nonlinearities may need to be introduced in the mathematical models involved in the simulation, leading to important and often overwhelming computational effort. This results in a huge number of degrees of freedom, a drawback for such complex models, so that they cannot be tackled with classical brute force methods. Therefore, alternative numerical approaches are required, such as model reduction methods. These methods exploit the fact that the response of complex models can often be approximated with a reasonable accuracy by the response of a surrogate model, seen as the projection of the initial one on a lower dimensional

functional basis [1–3]. Model reduction methods distinguish themselves by the way of defining and constructing the reduced basis. Among them are the Proper orthogonal decomposition (POD) [4, 5] and the Reduced Basis Method (RB) [6, 7]. While the first one seeks to extract the most relevant information from an already partial solution, the second one selects the most relevant calculation to perform in order to enrich the basis, thanks to an error indicator.

A third and promising model order reduction method is the Proper Generalized Decomposition (PGD) [8, 9] which was introduced as radial loading approximation in [10]. The PGD approximation does not require any knowledge of the solution (it is thus referred as *a priori*) nor orthogonality property. Based on a separated variable representation of the solution field, it operates in an iterative strategy in which a set of simple problems, that can be seen as pseudo eigenvalue problems, need to be solved. For time-dependent partial differential equations (PDEs), a separated representation of the solution  $u$  over the space-time domain is taken under the form of a sum of  $m$  products of space and time functions:

$$u(x, t) \simeq u_m(x, t) = \sum_{i=1}^m \psi_i(x) \lambda_i(t) \quad (1)$$

PGD has been developed during years for solving time-dependent nonlinear problems in Structural Mechanics [10–13] as a key-point of the so-called LATIN method. Extension to stochastic problems, initially under the name Generalized Spectral Decomposition has been done in [14]. The PGD approach has also led to major breakthroughs for real-time simulation [15], decision making tools [16], as well as multidimensional [17, 18] or multiphysics [12] problems.

Even though PGD is usually very effective, a major question is to derive verification tools for controlling the calculation process. Basic results on *a priori* error estimation for separation of variable representation can be found in [11], and a first work providing strict bounds on global error in the PGD context is [19], in which specific error indicators are also given.

The first goal of this paper is to compare the effectivity of the classical PGD computational methods which are all built on a progressive algorithm [20], rather than solving the  $m$  basis functions at once. A particular attention is paid to the numerical strategy developed around PGD methods since the beginning of the 90's [11]: for a time-space problem, a new PGD mode is computed using very few subiterations and at the end the full PGD time function basis is updated at once. In the literature, several variants of PGD have been proposed for the progressive construction of (1), respectively based on Galerkin formulation, Minimal Residual formulation [21] or Petrov–Galerkin formulation [22]. Performances of all these methods are compared on a benchmark problem proposed by S. Idelsohn, a one dimensional transient thermal problem with a moving *thermal load*. Secondly, we propose a new computational technique based on a PGD error indicator introduced in [23], minimizing the Constitutive Relation Error (CRE), with very promising properties. Finally, the effectivity of the variable separation hypothesis is analyzed with respect to the velocity of the moving thermal loading. In

this context, a lifting PGD computational method is proposed to push the limits of that hypothesis.

The paper outline is as follows: in “[The benchmark problem](#)” we present the one dimensional transient thermal benchmark problem. In “[Optimal reduced solution](#)”, an optimal reduced solution, computed from a brute force finite element solution, is proposed through the Singular Value Decomposition (SVD). Classical definitions of PGD methods applied to the benchmark problem are reviewed in “[Classical Proper Generalized Decomposition methods](#)”. We then introduce in “[Minimisation of the Constitutive Relation Error](#)” a non classical definition of PGD, called CRE-PGD. In “[A non-separable lifting algorithm](#)”, the lifting technique is addressed to study limits of variable separation. In “[Results and discussion](#)”, the different methods presented in this article are compared on the benchmark problem.

## Methods

### The benchmark problem

We consider a transient problem defined on a one dimensional domain  $\Omega = [0, L]$ , with boundary  $\partial\Omega = \{0; L\}$ , over the time interval  $\mathcal{I} = [0, T]$ . It is assumed that a prescribed zero temperature is applied on  $\partial\Omega$  and that the domain is subjected to a time-dependent thermal loading that consists of a thermal source  $f_d(x, t)$  in  $\Omega$ . This loading can be seen as a moving flame/laser beam such that  $f_d(x, t) = \delta(x - vt)$ , a Dirac distribution, where  $v$  is the velocity of the flame/laser beam (see Figure 1).

For the sake of simplicity, the initial conditions are set to zero. The material that composes  $\Omega$  is assumed to be homogeneous and fully known where  $c$  (resp.  $\mu$ ) represents the heat capacity (resp. conductivity) of the material.

The associated mathematical problem consists of finding the temperature-flux pair  $(u(x, t), \varphi(x, t))$ , with  $(x, t) \in \Omega \times \mathcal{I}$ , that verifies:

- the thermal constraints:

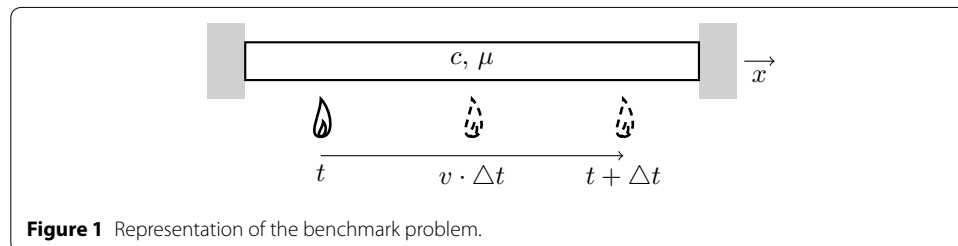
$$u(0, t) = u(L, t) = 0 \quad \forall t \in \mathcal{I} \quad (2a)$$

- the equilibrium equation:

$$c \frac{\partial u}{\partial t}(x, t) + \frac{\partial \varphi}{\partial x}(x, t) = \delta(x - vt) \quad \forall (x, t) \in \Omega \times \mathcal{I} \quad (2b)$$

- the constitutive relation:

$$\varphi(x, t) = -\mu \frac{\partial u}{\partial x}(x, t) \quad \forall (x, t) \in \Omega \times \mathcal{I} \quad (2c)$$



- the initial condition:

$$u(x, 0) = 0 \quad \forall x \in \Omega \quad (2d)$$

In the following, in order to be consistent with other linear problems encountered in Mechanics (linear elasticity for instance), we reform the variable  $\varphi \rightarrow -\varphi$  which leads, in particular, to the new constitutive relation  $\varphi = \mu \frac{\partial u}{\partial x}$ .

Defining  $\mathcal{V} = H_0^1(\Omega) = \{w \in H^1(\Omega), w|_{\partial\Omega} = 0\}$ , the weak formulation in space of the previous problem reads for all  $t \in \mathcal{I}$ :

$$\text{Find } u(x, t) \in \mathcal{V} \text{ such that } b(u, w) = l(w) \quad \forall w \in \mathcal{V} \quad (3)$$

with  $u|_{t=0} = 0$ . Bilinear form  $b(\cdot, \cdot)$  and linear form  $l(\cdot)$  are defined as:

$$b(u, w) = \int_{\Omega} \left\{ c \frac{\partial u}{\partial t} w + \mu \frac{\partial u}{\partial x} \frac{\partial w}{\partial x} \right\} dx, \quad l(w) = \int_{\Omega} \delta(x - vt) w dx$$

As regards the full weak formulation, functional spaces  $\mathcal{T} = L^2(\mathcal{I})$  and  $L^2(\mathcal{I}; \mathcal{V}) = \mathcal{V} \otimes \mathcal{T}$  are introduced. The solution  $u \in L^2(\mathcal{I}; \mathcal{V})$  is therefore searched, with  $\frac{\partial u}{\partial t} \in L^2(\mathcal{I}; L^2(\Omega))$ , such that:

$$B(u, w) = L(w) \quad \forall w \in L^2(\mathcal{I}; \mathcal{V}) \quad (4)$$

with

$$B(u, w) = \int_{\mathcal{I}} b(u, w) dt + \int_{\Omega} u(x, 0^+) w(x, 0^+) dx, \quad L(w) = \int_{\mathcal{I}} l(w) dt$$

The exact solution of (4), which is usually out of reach, is denoted  $u_{ex}$  (and  $\varphi_{ex} = \mu \frac{\partial u_{ex}}{\partial x}$ ). It may be approximated using the finite element method (FEM), considering time and space discretizations. Using the classical Galerkin approximation, approximate spaces are introduced:  $\mathcal{V}_h \subset \mathcal{V}$  (resp.  $\mathcal{T}_h \subset \mathcal{T}$ ) made from first order polynomials basis of dimension  $N_x$  (resp.  $N_t$ ). The resulting  $N_x \times N_t$  linear system leads to a costly brute-force evaluation when  $N_x$  and/or  $N_t$  are large. Such a solution can be represented as a  $N_x \times N_t$  matrix of the following discrete values:

$$\mathbb{U} = \begin{pmatrix} u_h(x_1, t_1) & u_h(x_1, t_2) & \cdots & u_h(x_1, t_{N_t}) \\ u_h(x_2, t_1) & u_h(x_2, t_2) & \cdots & u_h(x_2, t_{N_t}) \\ \vdots & & \ddots & \vdots \\ u_h(x_{N_x}, t_1) & \cdots & \cdots & u_h(x_{N_x}, t_{N_t}) \end{pmatrix}$$

### Optimal reduced solution

Let us suppose that the full discrete approximated solution  $u_h$  is known (under the form  $\mathbb{U}$ ), and one looks to extract basis functions such as  $u_h$  can be approximated under the separated representation form (1).

The problem consists of defining the separated representation  $u_m$  (1) of the solution  $u_h$  of (3) as the one which minimizes the distance to the approximated solution  $u_h$  with respect to a given norm  $\|\cdot\|$  on  $\mathcal{V} \otimes \mathcal{T}$ .

$$\|u_h(x, t) - u_m(x, t)\|^2 = \min_{\psi_i, \lambda_i} \left\| u_h(x, t) - \sum_{i=1}^m \psi_i(x) \lambda_i(t) \right\|^2 \quad (5)$$

where functions  $\psi_i$  and  $\lambda_i$  are the optimal reduced basis functions with respect to the chosen norm.

The SVD is a method that transforms correlated data into uncorrelated ones, identifies and orders the dimensions along which data points exhibit the highest variations. From a mathematical point of view, this method is just a transformation which diagonalizes a given matrix  $\mathbb{A}$  of size  $M \times N$  and brings it to a canonical form [24]:

$$\mathbb{A} = \mathbb{W} \Sigma \mathbb{V}^\dagger$$

where<sup>a</sup>  $\mathbb{W}$  (resp.  $\mathbb{V}$ ) is a unitary matrix of size  $M \times M$  (resp.  $N \times N$ ) consisting of the  $r$  first right (resp. left) singular vectors,  $r$  being the rank of  $\mathbb{A}$ , and  $\Sigma$  is a diagonal matrix formed by the singular values  $\sigma_i$  of  $\mathbb{A}$  such that  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0$ .

Since singular values are arranged in a specific order, a best approximation  $\mathbb{A}_k$  of the original data points  $\mathbb{A}$  using fewer dimensions can be determined by simply ignoring singular values below a particular threshold ( $k \leq r$ ) to massively reduce the data while preserving the main behavior of  $\mathbb{A}$ . The implied matrix  $\mathbb{A}_k$  of rank  $k$  verifies the Eckart–Young theorem according to the Euclidean norm  $\|\cdot\|_2$  [25]:

$$\min_{\text{rank}(X) \leq k} \|\mathbb{A} - X\|_2 = \|\mathbb{A} - \mathbb{A}_k\|_2 = \sigma_{k+1} \quad (6)$$

Therefore, if the singular values decreases rapidly, a good approximation of  $\mathbb{A}$  with small rank can be found.

For physical problems, the energy norm  $\|\cdot\|_E$  is the natural norm in which measuring the error between  $u_{ex}$  and  $u_h$ . Therefore, to compare SVD and PGD algorithms, we choose the associated energy scalar product, denoted  $\langle\langle \cdot, \cdot \rangle\rangle_E$ , that verifies the variable separation property:

$$\langle\langle u, w \rangle\rangle_E = \left\langle \sqrt{\langle u(t), w(t) \rangle_{\mathcal{V}}}, \sqrt{\langle u(t), w(t) \rangle_{\mathcal{V}}} \right\rangle_{\mathcal{T}} \quad (7a)$$

$$\langle u(x, t), w(x, t) \rangle_{\mathcal{T}} = \int_{\mathcal{I}} u(x, t) w(x, t) dt \quad (7b)$$

$$\langle u(x, t), w(x, t) \rangle_{\mathcal{V}} = \int_{\Omega} \frac{\partial u}{\partial x}(x, t) \frac{1}{\mu} \frac{\partial w}{\partial x}(x, t) dx \quad (7c)$$

The implied discrete inner products  $\langle \cdot, \cdot \rangle_{\mathcal{V}_h}$  and  $\langle \cdot, \cdot \rangle_{\mathcal{T}_h}$  can be written:

$$\langle u, w \rangle_{\mathcal{V}_h} = \mathbf{u}^\dagger \mathbb{K} \mathbf{w}, \quad \langle u, w \rangle_{\mathcal{T}_h} = \mathbf{u} \mathbb{M} \mathbf{w}^\dagger$$

where  $\mathbb{K} \in \mathbb{R}^{N_x^2}$  (resp.  $\mathbb{M} \in \mathbb{R}^{N_t^2}$ ) is the finite element stiffness matrix (resp. the finite element mass matrix in time) and  $\mathbf{u}, \mathbf{w} \in \mathbb{R}^{N_x \times N_t}$  the FEM solution. Employing a Cholesky

factorization, the inner product  $\langle \cdot, \cdot \rangle_E$  can be transformed to the standard Euclidean inner product as:

$$\|u\|_{\mathcal{V}_h} = \|\sqrt{\mathbb{K}}u\|_2, \quad \|u\|_{\mathcal{T}_h} = \|u\sqrt{\mathbb{M}}\|_2 \quad (8)$$

To avoid the use of the Euclidean norm in (6) and thanks to the transformation (8), ones can compute the reduced SVD according to the energy norm through the regular SVD algorithm on the matrix  $\sqrt{\mathbb{K}}U\sqrt{\mathbb{M}}$  rather than on the matrix  $U$ . Therefore, the separated variable decomposition  $U = \Psi\Lambda^\dagger$  obtained from the regular SVD  $W\Sigma V^\dagger$  verifies:

$$\begin{aligned} \sqrt{\mathbb{K}}\Lambda\sqrt{\mathbb{M}} \left( \sqrt{\mathbb{K}}\Lambda\sqrt{\mathbb{M}} \right)^\dagger &= \sqrt{\mathbb{K}}\Lambda\Lambda^\dagger\sqrt{\mathbb{K}}^\dagger = \sqrt{\mathbb{K}}\Psi\Psi^\dagger\sqrt{\mathbb{K}}^\dagger = W\Sigma^2W^\dagger \\ \left( \sqrt{\mathbb{K}}\Lambda\sqrt{\mathbb{M}} \right)^\dagger \sqrt{\mathbb{K}}\Lambda\sqrt{\mathbb{M}} &= \sqrt{\mathbb{M}}^\dagger\Lambda^\dagger\Lambda\sqrt{\mathbb{M}} = \sqrt{\mathbb{M}}^\dagger\Lambda^\dagger\Lambda\sqrt{\mathbb{M}} = V\Sigma^2V^\dagger \end{aligned}$$

Matrices  $\Psi$  and  $\Lambda$ , that contain the reduced basis functions  $\psi_i$  and  $\lambda_i$  introduced in (1), have to verify the following relations:  $\Psi = \sqrt{\mathbb{K}}^{-1}U$ ,  $\Lambda = \Sigma \left( \sqrt{\mathbb{M}_t}^{-1} \right)^\dagger V$

*Remark 1* The reduced basis functions  $\psi_i \in \mathcal{V}_h$  (resp.  $\lambda_i \in \mathcal{T}_h$ ) are orthogonal with respect to the inner product  $\langle \cdot, \cdot \rangle_{\mathcal{V}_h}$  (resp.  $\langle \cdot, \cdot \rangle_{\mathcal{T}_h}$ ).

### Classical PGD methods

Usually, the brute force solution  $u_h$  is out of reach. In this section, classical PGD strategies are reviewed, aiming to approximate the solution  $u$  under the form  $u_m$  that verifies (1) without knowledge of the solution  $u$ . In these methods, the reduced basis is computed on the fly as the problem is solved.

Rather than solving the  $m$  basis functions at once, which is not practical for large scale applications because of prohibitive computational cost, PGD algorithms are built on a progressive algorithm [20]. Assuming that a decomposition  $u_m$  is known (previously computed), a new couple  $(\psi, \lambda) \in \mathcal{V} \times \mathcal{T}$  is searched such that  $u$  is of the form (PGD decomposition of order  $m + 1$ ):

$$u(x, t) \simeq u_{m+1}(x, t) = u_m(x, t) + \psi(x)\lambda(t) \quad (9)$$

In the following and for clarity reasons, the functions dependency is no longer indicated.

### Galerkin PGD

First, a PGD computational method based on Galerkin orthogonality [11] and well-suited for diffusion-dominated problems is presented. By injecting the separated variable form (9) in (4), the new couple  $(\psi, \lambda) \in \mathcal{V} \times \mathcal{T}$  is the optimal one that verifies the Galerkin orthogonality:

$$B(u_m + \psi\lambda, \psi^*\lambda + \psi\lambda^*) = L(\psi^*\lambda + \psi\lambda^*) \quad \forall \psi^* \in \mathcal{V}, \lambda^* \in \mathcal{T} \quad (10)$$

That equation naturally leads to the following problems :

- the weak formulation of a partial differential equation (PDE) in space, usually approximated by FEM:

$$B(u_m + \psi \lambda, \psi^* \lambda) = L(\psi^* \lambda) \quad \forall \psi^* \in \mathcal{V} \quad (11)$$

- the weak formulation of ordinary differential equation (ODE) in time:

$$B(u_m + \psi \lambda, \psi \lambda^*) = L(\psi \lambda^*) \quad \forall \lambda^* \in \mathcal{T}. \quad (12)$$

A couple  $(\psi, \lambda) \in \mathcal{V} \times \mathcal{T}$  then verifies (10) if and only if  $\psi$  verifies (11) and  $\lambda$  verifies (12), which is a non-linear problem. As this problem can be interpreted as a pseudo-eigenproblem [22], a natural algorithm to capture the dominant eigenfunction consists of a power iterations strategy. Starting from an initial random time function  $\lambda$  (verifying initial conditions), each iteration of the power algorithm verifies a sequence of lower dimensional problems (11–12), leading to Algorithm 1.

*Remark 2* The key property of Galerkin computational method is that the error is orthogonal to the approximation spaces. Therefore,  $u_{ex} - u_m$  is orthogonal to  $\mathcal{V}_h \otimes \mathcal{T}_h$  according to  $B(\cdot, \cdot)$  scalar product.

```

for  $m = 1$  to  $m_{max}$  do
  Initialize  $\lambda$ 
  for  $k = 1$  to  $k_{max}$  do
    Compute  $\psi$  from (11)
    Normalize  $\psi$ 
    Compute  $\lambda$  from (12)
  end
  Set  $\psi_m = \psi$  and  $\lambda_m = \lambda$ 
  Set  $u_m = u_{m-1} + \psi_m \lambda_m$  and check convergence
end

```

**Algorithm 1:** Galerkin PGD

*Remark 3* Normalization of space  $\psi$  or time  $\lambda$  functions is preferable for stability reason of the power iterations algorithm. In Algorithm 1, we arbitrarily choose to normalize the space function  $\psi$ .

*Remark 4* One could check the convergence of  $\psi \lambda$  to stop the power iterations. As explained in [22], a coarse criterion is sufficient to obtain good approximation as convergence is reached quickly. In practice, we rather prefer to fix a number of sub-iterations  $k_{max} = 4$ , letting the next PGD mode to correct the previous one if necessary.

#### Improvement of the decomposition

A post-processing of the computed functions can be made to improve the decomposition, like orthogonalization of space basis functions. A simple modification to the power iterations algorithm consists in updating the whole set of time functions  $\Lambda_m = \{\lambda_i\}_{i=1\dots m}$  after every  $n$ -th new PGD mode. These functions are defined by a system of ODEs of dimension  $m$  analogous to (12). A mapping application  $T$  is defined taking the whole set of space functions  $\Psi_m = \{\psi\}_{i=1\dots m} \in \mathcal{V}^m$  into time functions  $\Lambda_m = T(\Psi_m) \in \mathcal{T}^m$  defined by:

$$B(\Psi_m \cdot \Lambda_m, \Psi_m \cdot \Lambda_m^*) = L(\Psi_m \cdot \Lambda_m^*) \quad \forall \Lambda_m^* \in \mathcal{T}^m \quad (13)$$

This post-processing improves the quality of the progressive PGD algorithm, although the obtained decomposition is not the optimal one. If the update step is done after each new PGD mode, it is added after the power iterations step, as shown in Algorithm 2. Known as update step [26], it can be applied to all PGD computational methods.

```

for  $m = 1$  to  $m_{max}$  do
  Initialization
  for  $k = 1$  to  $k_{max}$  do
    | Power iterations step
  end
  Set  $\psi_m = \psi$ ,  $\lambda_m = \lambda$ 
  Compute  $\Lambda_m = T(\Psi_m)$ 
  Set  $u_m = \Psi_m \cdot \Lambda_m$  and check convergence
end

```

**Algorithm 2:** The update step

### Minimal residual PGD

The next technique is a PGD strategy based on a minimal residual criterion [21]. This construction presents monotonic convergence of the decomposition in the sense of residual norm.

The residual  $\mathcal{R}(u) \in \mathcal{V} \otimes \mathcal{T}$  is defined under the form:

$$\langle\langle w, \mathcal{R}(u) \rangle\rangle = L(w) - B(u, w) = \langle\langle w, \delta - \underbrace{\left( k \frac{\partial u}{\partial x} + c \frac{\partial u}{\partial t} \right)}_{B(u)} \rangle\rangle$$

Therefore, an optimal couple  $(\psi, \lambda) \in \mathcal{V} \times \mathcal{T}$  is defined as the one that minimizes the residual norm:

$$(\psi, \lambda) = \arg \min_{\psi, \lambda} \|\mathcal{R}(u_m + \psi \lambda)\|^2$$

or equivalently:

$$(\psi, \lambda) = \arg \min_{\psi, \lambda} \frac{1}{2} \langle\langle \mathcal{B}(\psi \lambda), \mathcal{B}(\psi \lambda) \rangle\rangle - \langle\langle \mathcal{R}(u_m), \mathcal{B}(\psi \lambda) \rangle\rangle \quad (14)$$

In a continuous framework, (14) leads to a non-classical formulation as it requires the introduction of a more refined functional space of the space domain  $\mathcal{V}$  in order to guaranty existence and uniqueness of solutions (by introducing  $H^2(\Omega) \otimes H^1(\mathcal{T})$  in place of  $H^1(\Omega) \otimes L^2(\mathcal{T})$  for our benchmark problem). In practice, a minimal residual formulation is applied to the discretized problem (4) after introducing approximation spaces  $\mathcal{V}_h$  and  $\mathcal{T}_h$ .

Therefore, by introducing the discrete  $N_x N_t \times N_x N_t$  matrix  $\mathbb{B}$  (resp.  $N_x N_t$  vector  $\mathbf{L}$ ) of the bilinear operator  $B(\cdot, \cdot)$  (resp. the linear operator  $L(\cdot)$ ), the discrete  $N_x N_t$  residual vector  $\mathbf{R}$  of  $\mathcal{R}(\cdot)$  takes the form:

$$\mathbf{R}(u) = \mathbb{B}u - \mathbf{L}, \quad u = \begin{bmatrix} u(t_1) \\ u(t_2) \\ \vdots \end{bmatrix}$$



such that an optimal couple  $(\psi, \lambda) \in \mathcal{V}_h \times \mathcal{T}_h$  minimizes the discretized residual norm (Euclidean norm, since the matrix  $\mathbb{B}$  is not invertible):

$$(\psi, \lambda) = \arg \min_{\psi, \lambda} \|\mathbf{R}(u_m + \psi \lambda)\|_2^2$$

By introducing notation  $\mathbf{u}_{m+1} = \mathbf{u}_m + \psi \cdot \lambda$ , the minimization leads to the problem:

$$(\psi^* \cdot \lambda + \psi \cdot \lambda^*)^T \left[ \mathbb{B}^T \mathbb{B}(\mathbf{u}_m + \psi \cdot \lambda) - \mathbb{B}^T \mathbf{L} \right] = 0 \quad (15)$$

Let us define the  $N_x \times N_x$  matrices  $\mathbb{C}_{ij}$  and the  $N_x$  vectors  $\mathbf{F}_i$  such that:

$$\mathbb{B}^T \mathbb{B} = \begin{bmatrix} \mathbb{C}_{11} & \mathbb{C}_{12} & \cdots \\ \mathbb{C}_{21} & \ddots & \\ \vdots & & \ddots \end{bmatrix}, \quad \mathbb{B}^T \mathbf{L} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \\ \vdots \end{bmatrix}$$

The discrete minimization (15) leads to the following reduced problems:

- a discretized space problem:

$$\left( \sum_{i,j} \lambda_i \lambda_j \mathbb{C}_{ij} \right) \psi = \sum_i \lambda_i \mathbf{F}_i \quad (16)$$

- a discretized time problem:

$$\begin{bmatrix} \psi^T \mathbb{C}_{11} \psi & \psi^T \mathbb{C}_{12} \psi & \cdots \\ \psi^T \mathbb{C}_{21} \psi & \ddots & \\ \vdots & & \ddots \end{bmatrix} \lambda = \begin{bmatrix} \psi^T \mathbf{F}_1 \\ \psi^T \mathbf{F}_2 \\ \vdots \end{bmatrix} \quad (17)$$

*Remark 5* As for all Galerkin based methods, the error  $u_{ex} - u_m$  is orthogonal to the approximation space  $\mathbb{B}^T \mathbb{B}(\mathcal{V}_h \otimes \mathcal{T}_h)$  according to the Euclidean scalar product, and depends strongly on the conditioning of the operator  $\mathbb{B}^T \mathbb{B}$ .

```

for  $m = 1$  to  $m_{max}$  do
  Initialize  $\lambda$ 
  for  $k = 1$  to  $k_{max}$  do
    Compute  $\psi$  from (16)
    Normalize  $\psi$ 
    Compute  $\lambda$  from (17)
  end
  Set  $\psi_m = \psi$  and  $\lambda_m = \lambda$ 
  Set  $u_m = u_{m-1} + \psi_m \lambda_m$  and check convergence
end

```

**Algorithm 3:** Minimal Residual PGD

### Petrov–Galerkin PGD

In this section, another possible PGD computational method is presented, based on a Petrov–Galerkin criterion, also called MinMax [22]. Such a formulation is frequently used for solving PDEs which contain terms with odd order, implying a loss of symmetry in the weak formulation, as transport-dominated problems.

This Petrov–Galerkin formulation uses the weak formulation (4) where the unknown  $u$  and test function  $w$  are in different finite dimensional subspaces. Under the variable separation assumption, the test function  $w$  is taken as another couple of space and time functions  $(\tilde{\psi}, \tilde{\lambda}) \in \mathcal{V} \times \mathcal{T}$  such that the new PGD couple  $(\psi, \lambda) \in \mathcal{V} \times \mathcal{T}$  is the optimal one that verifies the Galerkin orthogonality:

$$B(u_m + \psi \lambda, \tilde{\psi}^* \tilde{\lambda} + \tilde{\psi} \tilde{\lambda}^*) = L(\tilde{\psi}^* \tilde{\lambda} + \tilde{\psi} \tilde{\lambda}^*) \quad \forall \tilde{\psi}^* \in \mathcal{V}, \tilde{\lambda}^* \in \mathcal{T} \quad (18)$$

This leads to the two following orthogonality criteria:

$$B(u_m + \psi \lambda, \tilde{\psi}^* \tilde{\lambda}) = L(\tilde{\psi}^* \tilde{\lambda}) \quad \forall \tilde{\psi}^* \in \mathcal{V} \quad (19a)$$

$$B(u_m + \psi \lambda, \tilde{\psi} \tilde{\lambda}^*) = L(\tilde{\psi} \tilde{\lambda}^*) \quad \forall \tilde{\lambda}^* \in \mathcal{T} \quad (19b)$$

As a matter of fact, additional equations must be added in order to define the new functions  $(\tilde{\psi}, \tilde{\lambda}) \in \mathcal{V} \times \mathcal{T}$ . The following orthogonality criteria (20a), (20b) are used, where  $\langle \cdot, \cdot \rangle$  is an inner product on  $\mathcal{V} \otimes \mathcal{T}$ . To compare PGD computational methods one another, we select the energy inner product (7a)–(7c).

$$B(\psi^* \lambda, \tilde{\psi} \tilde{\lambda}) = \langle \langle \psi^* \lambda, \psi \lambda \rangle \rangle_E \quad \forall \psi^* \in \mathcal{V} \quad (20a)$$

$$B(\psi \lambda^*, \tilde{\psi} \tilde{\lambda}) = \langle \langle \psi \lambda^*, \psi \lambda \rangle \rangle_E \quad \forall \lambda^* \in \mathcal{T} \quad (20b)$$

As for previous algorithms, an approximation of  $(\psi, \tilde{\psi}, \lambda, \tilde{\lambda}) \in \mathcal{V}^2 \times \mathcal{T}^2$  is computed to verify (18) and (20a), (20b) simultaneously. As such a problem is non-linear, a power iterations strategy is chosen, where the four lower dimension problems are solved iteratively one after the other (see Algorithm 4).

**Remark 6** As for all Galerkin based methods, the error  $u_{ex} - u_m$  is orthogonal to an approximation space  $\mathcal{L} = \left\{ \tilde{\psi}_i \otimes \tilde{\lambda}_i \right\}_{i=1 \dots m}$  of the test space according to the  $B(\cdot, \cdot)$  scalar product.

```

for  $m = 1$  to  $m_{max}$  do
  Initialize  $\lambda, \tilde{\lambda}$ 
  for  $k = 1$  to  $k_{max}$  do
    Compute  $\psi$  from (19a)
    Normalize  $\psi$ 
    Compute  $\tilde{\psi}$  from (20a)
    Normalize  $\tilde{\psi}$ 
    Compute  $\lambda$  from (19b)
    Compute  $\tilde{\lambda}$  from (20b)
  end
  Set  $\psi_m = \psi$  and  $\lambda_m = \lambda$ 
  Set  $u_m = u_{m-1} + \psi_m \lambda_m$  and check convergence
end

```

**Algorithm 4:** Petrov-Galerkin PGD**Minimisation of the CRE**

To control the PGD computational process, specific error indicators have been built in recent years. They particularly assess the error due to truncation of the sum in the separated decomposition (1). A first robust approach for PGD verification, using the concept of CRE, was proposed in [27, 28] leading to guaranteed and relevant error evaluation. Its specificity lies on the way to construct the required dual fields, as  $\varphi(u_m) = \mu \frac{\partial u_m}{\partial x}$  does not verify equilibrium (3). The solution leads to a variable separation of  $\varphi(u_m)$  with a static formulation, done from and after the classical Galerkin PGD space problem. The obtained data combine with prescribed ones, as displacements, traction and body forces define the starting point to evaluate the reduction error.

In this error estimator, the required dual field is computed after the PGD procedure and then cannot influence the solution. Here we propose to compute and control the PGD procedure on the fly with the CRE estimator.

The equilibrium is reformulated by introducing flux fields  $\tau$  and  $Q$  such that:

$$\int_{\mathcal{I}} \int_{\Omega} c \frac{\partial u}{\partial t} w + \underbrace{(\varphi - Q)}_{\tau} \frac{\partial w}{\partial x} dx dt = 0 \quad \forall w \in \mathcal{V} \otimes \mathcal{T} \quad (21)$$

We wish to minimize the CRE estimator (22), with the constraint that the flux  $\tau$  verifies equilibrium (21), with  $u$  a kinematically admissible field written with separated representation form.

$$(u(x, t), \tau(x, t)) = \arg \min_{u, \tau} \int_{\mathcal{I}} \int_{\Omega} \frac{1}{\mu} \left[ \tau(x, t) + Q(x, t) - \mu \frac{\partial u}{\partial x}(x, t) \right]^2 dx dt \quad (22)$$

As the flux  $\tau$  has to verify the equilibrium (21),  $\tau$  can also be written under the variable separation form:

$$\tau(x, t) \simeq \tau_m(x, t) = \sum_{i=1}^m \frac{d\lambda_i}{dt}(t) Z_i(x) \quad (23)$$

Indeed, by injecting assumption (1) into equilibrium (21) with a progressive algorithm solver,  $\tau$  has to verify, at each PGD stage  $m$ , the following equation:

$$\forall k, \forall t, \quad \int_{\Omega} \sum_{i=1}^k \left[ c \frac{d\lambda_i}{dt} \psi_i \right] w + \tau \frac{\partial w}{\partial x} dx = 0, \quad \forall w \in \mathcal{V} \quad (24)$$

The chosen separated form (23) leads to an interesting condition between  $Z_i$  and  $\psi_i$ :

$$\forall i, \forall t, \quad \int_{\Omega} Z_i(x) \frac{\partial w}{\partial x} + c \psi_i(x) w dx = 0, \quad \forall w \in \mathcal{V} \quad (25)$$

Therefore, the progressive algorithm adopted in “[Classical Proper Generalized Decomposition methods](#)” also applies on  $\tau$ . It assumes that decomposition  $u_m$  and  $\tau_m$  are known (previously computed) and it looks for a new couple  $(\psi, \lambda, Z)$  such that  $u$  verifies (9) and  $\tau$  verifies:

$$\tau(x, t) \simeq \tau_{m+1}(x, t) = \tau_m(x, t) + \frac{d\lambda}{dt}(t) Z(x) \quad (26)$$

#### A full minimization

Minimizing the CRE (22) under conditions (25) leads to minimizing the following equation for  $\psi \in \mathcal{V}, \lambda \in \mathcal{T}$  and  $Z$  where the constraint is taken into account through Lagrange multipliers:

$$\begin{aligned} (\psi, \lambda, Z, w) = \arg \min_{\psi, \lambda, Z} \max_w \int_{\mathcal{I}} \int_{\Omega} \frac{1}{\mu} \left( Q + \tau_m - \mu \frac{\partial u_m}{\partial x} + \frac{d\lambda}{dt} Z - \mu \frac{d\psi}{dx} \lambda \right)^2 dx dt \\ + \int_{\Omega} Z \frac{dw}{dx} + c \psi w dx \end{aligned} \quad (27)$$

Solving the saddle point problem (27) according to each unknown leads to:

- According to  $\lambda$ :

$$\begin{aligned} 0 = \int_{\mathcal{I}} \int_{\Omega} \frac{1}{\mu} \left( Q + \tau_m - \mu \frac{\partial u_m}{\partial x} + \frac{d\lambda}{dt} Z - \mu \lambda \frac{d\psi}{dx} \right) \left( \frac{d\lambda^*}{dt} Z - \mu \lambda^* \frac{d\psi}{dx} \right) dx dt \\ \forall \lambda^* \in \mathcal{T} \end{aligned} \quad (28a)$$

- According to  $\psi$ :

$$\begin{aligned} 0 = 2 \int_{\mathcal{I}} \int_{\Omega} \left( \mu \frac{\partial u_m}{\partial x} - Q - \tau_m - \frac{d\lambda}{dt} Z + \mu \lambda \frac{d\psi}{dx} \right) \lambda \frac{d\psi^*}{dx} dx dt \\ + \int_{\Omega} c \psi^* w dx \\ \forall \psi^* \in \mathcal{V} \end{aligned} \quad (28b)$$

- According to  $Z$ :

$$0 = 2 \int_{\mathcal{T}} \int_{\Omega} \frac{1}{\mu} \left( Q + \tau_m - \mu \frac{\partial u_m}{\partial x} + \frac{d\lambda}{dt} Z - \mu \lambda \frac{d\psi}{dx} \right) \frac{d\lambda}{dt} Z^* dx dt + \int_{\Omega} Z^* \frac{\partial w}{\partial x} dx \quad \forall Z^* \in \mathcal{S} \quad (28c)$$

$$\text{where } \mathcal{S} = \left\{ w \in L^2(\Omega) \mid \forall g \in L^2(\Omega), \int_{\Omega} g \frac{dw}{dx} dx = 0 \right\}.$$

- According to  $w$ :

$$0 = \int_{\Omega} Z \frac{\partial w^*}{\partial x} + c\psi w^* dx \quad \forall w^* \in \mathcal{V}. \quad (28d)$$

As (28b)–(28d) are linked together, discrete finite element solutions  $(\psi_h, Z_h, w_h)$  are determined at once by inverting a matrix problem of thrice the problem size in space:

$$\mathbb{B} \begin{pmatrix} \psi_h \\ Z_h \\ w_h \end{pmatrix} = L$$

It should be noted that contrary to the usual case of CRE, discretization errors are not taken into account there. As a matter of fact, very refined space and time meshes will be taken to omit it.

#### A partial minimization

The full minimization leads to invert a huge problem. In the general case, it is computationally expensive. We propose here a partial minimization where condition (21) is not taken into account in the minimization process but afterwards. That leads to a new and promising PGD computational strategy.

Minimizing (22) under constraint (25) leads to the minimization in  $\mathcal{V} \otimes \mathcal{T} \otimes \mathcal{S}$  of:

$$(\psi, \lambda, Z) = \arg \min_{\psi, \lambda, Z} \int_{\mathcal{T}} \int_{\Omega} \frac{1}{\mu} \left( Q + \tau_m - \mu \frac{\partial u_m}{\partial x} + \frac{d\lambda}{dt} Z - \mu \frac{d\psi}{dx} \lambda \right)^2 dx dt \quad (29)$$

Minimization (29) leads to:

- According to  $\lambda$ :

$$0 = \int_{\mathcal{T}} \int_{\Omega} \mu \left( Q + \tau_m + \frac{\partial \lambda}{\partial t} Z - \frac{\partial u_m}{\partial x} - \frac{d\psi}{dx} \lambda \right) \left( Z \frac{d\lambda^*}{dt} - \mu \frac{d\psi}{dx} \lambda^* \right) dx dt \quad \forall \lambda^* \in \mathcal{T} \quad (30)$$

- According to  $\psi$ . The field  $Z$  is fixed to the previous value computed at the previous iteration of the power iterations algorithm in first place:

$$0 = \int_{\mathcal{I}} \int_{\Omega} \left( \frac{\partial u_m}{\partial x} + \frac{d\psi}{dx} \lambda - Q + \tau_m + \frac{\partial \lambda}{\partial t} Z \right) \lambda \frac{d\psi^*}{dx} dx dt \quad \forall \psi^* \in \mathcal{V} \quad (31)$$

- One can then determine  $Z$  to verify (25).

```

for  $m = 1$  to  $m_{max}$  do
  Initialize  $\lambda, \psi$ 
  for  $k = 1$  to  $k_{max}$  do
    Compute  $\lambda$  from (30)
    Compute  $\psi$  from (31)
    Compute  $Z$  from (25)
  end
  Set  $\psi_m = \psi, \lambda_m = \lambda$  and  $Z_m = Z$ 
  Set  $u_m = u_{m-1} + \psi_m \lambda_m$  and  $\tau_m = \tau_{m-1} + \frac{d\lambda_m}{dt} Z_m$  and check convergence
end

```

**Algorithm 5:** Partial Minimal CRE PGD

*Remark 7* The CRE quantity  $\varphi_m - \mu \frac{\partial u_m}{\partial x} = \tau_m + Q - \mu \frac{\partial u_m}{\partial x}$  is orthogonal to  $\mathcal{V} \otimes \mathcal{T}$  and  $\mathcal{S} \otimes \mathcal{T}$  approximation spaces according to the energy scalar product.

*Remark 8* For stationary problems, condition (24) does not apply. The idea is to seek  $\tau \in \mathcal{S}_0$  of the form:

$$\tau \simeq \tau_m = \sum_{i=1}^m \tilde{Z}_i(x, t)$$

where each  $\tilde{Z}_i \in \mathcal{S}_0 = \left\{ w \in L^2(\Omega, \mathcal{I}) \mid \forall z \in L^2(\Omega, \mathcal{I}), \int_{\mathcal{I}} \int_{\Omega} w \frac{\partial z}{\partial x} dx dt = 0 \right\}$ . Since potential and complementary energies are decoupled, each  $\tilde{Z}_i$  is decoupled from  $\psi_i$ .

### A non-separable lifting algorithm

In this section, a solution of the benchmark problem is computed under the sum of two contributions:

- an enrichment function, determined analytically in an infinite domain, which does not verify the separated decomposition;
- an additional function, that aims at verifying the boundary conditions, and which verifies variables separation.
- For the specific benchmark problem defined in (2a)–(2d), we look for the solution in an infinite domain without space boundary conditions. Let us set  $X = x - vt$  such that  $U(X) = U(x - vt) = u(x, t)$ . The problem then reads:

$$\begin{cases} cv \frac{dU}{dX} + \mu \frac{d^2 U}{dX^2} = -\delta(X) \\ U(x) = 0 \end{cases}$$

The solution of such an ODE is straightforward, leading to the following equation, where  $\theta$  stands for the Heaviside function and  $K$  is a real constant.

$$u_{\infty}(x, t, K) = \frac{\theta(x) + K}{cv} \left(1 - \exp^{-\frac{cv}{\mu}x}\right) - \frac{\theta(x - vt) + K}{cv} \left(1 - \exp^{-\frac{cv}{\mu}(x-vt)}\right)$$

Therefore, a dedicated algorithm is proposed, called Lifting PGD, where the solution is computed with one of the previous PGD algorithms initialized by  $u_{\infty}$  with  $K = 0$ , i.e. removing the homogeneous solution. To respect boundary conditions in space,  $u_{\infty}$  is multiplied with ad-hoc functions. As the solution  $u$  is equal to  $u_{\infty}$  far from the boundary conditions,  $u$  is searched under a specific form to correct  $u_{\infty}$  near the boundary conditions.

$$u(x, t) \approx u_{\infty}(x, t, 0) \times \delta(x) \times \delta(L - x) + u_m(x, t)$$

To compute the space and time basis functions  $\psi_i$  and  $\lambda_i$ , one can select one of the previously presented PGD strategies.

## Results and discussion

On the benchmark problem presented in “[The benchmark problem](#)”, we compare the behavior of the different PGD strategies detailed in this paper. The comparison focuses on: (i) quality of approximation by studying the convergence of the Greedy algorithm; (ii) computational cost. Finally, the influence of time effects is studied through the loading velocity parameter, to show the main limitations of PGD.

The reference solution  $u$  is the classical Galerkin approximation  $u_p$  solution of (3), obtained through a brute force FEM. To avoid influence of discretization error, very refined space and time meshes are chosen.

After defining some error indicators, performances of the different PGD strategies are conducted on the benchmark problem with  $L = 3.14m$ ,  $T = 1s$ ,  $c = 0.1/m^{-3}K^{-1}$ ,  $\mu = 1Wm^{-1}K^{-1}$ ,  $\nu = \frac{L}{2T}$  to form a reasonable test case. Later in “[Limits of the Proper Generalized Decomposition](#)”, the influence of these coefficient values will be studied to illustrate limits of the PGD method.

## Error estimation

To evaluate the quality of approximation, some error indicators are defined:

### Reference reduction error

Since the reference solution is known, an exact reduction error can be defined as the distance between the reference  $u_h$  and the reduced solution  $u_m$  according to the energy norm. As the mesh is highly refined, this reduction error is the exact one:

$$e_{ref} = \|u_h - u_m\|_E, \quad \bar{e}_{ref} = \frac{e_{ref}}{\|u_h\|_E}$$

### Residual error estimator

In general, the reference solution  $u_h$  is not at hand. Using the equilibrium defects of the approximate solution  $u_m$ , an approximated measure of the real error can be performed without knowledge of the reference solution.

$$e_{res} = \|\mathcal{R}(u_m)\|_{L^2}, \quad \bar{e}_{res} = \frac{e_{res}}{\|\delta\|_{L^2}}$$

### Constitutive Relation Error estimator

As explained in “Minimisation of the Constitutive Relation Error”, minimizing the CRE leads to an immediate error estimator, evaluated without knowledge of the reference solution, as:

$$e_{CRE} = \|\tau_m - \mu \frac{\partial u_m}{\partial x}\|_E, \quad \overline{e_{CRE}} = \frac{e_{CRE}}{\|\tau_m + \mu \frac{\partial u_m}{\partial x}\|_E}$$

As the exact reduction error is available, one can also define an effectivity index  $\eta$  in order to quantify the quality of the error estimators. Due to the use of a  $L^2$  norm to evaluate the residual error estimation, we only define the effectivity for the CRE estimator as:

$$\eta = \frac{e_{CRE}}{e_{ref}}$$

An error estimator is (i) even more relevant and accurate as the value of its associated effectivity index  $\eta$  is close to 1; (ii) said guaranteed when the value of  $\eta$  is above 1.

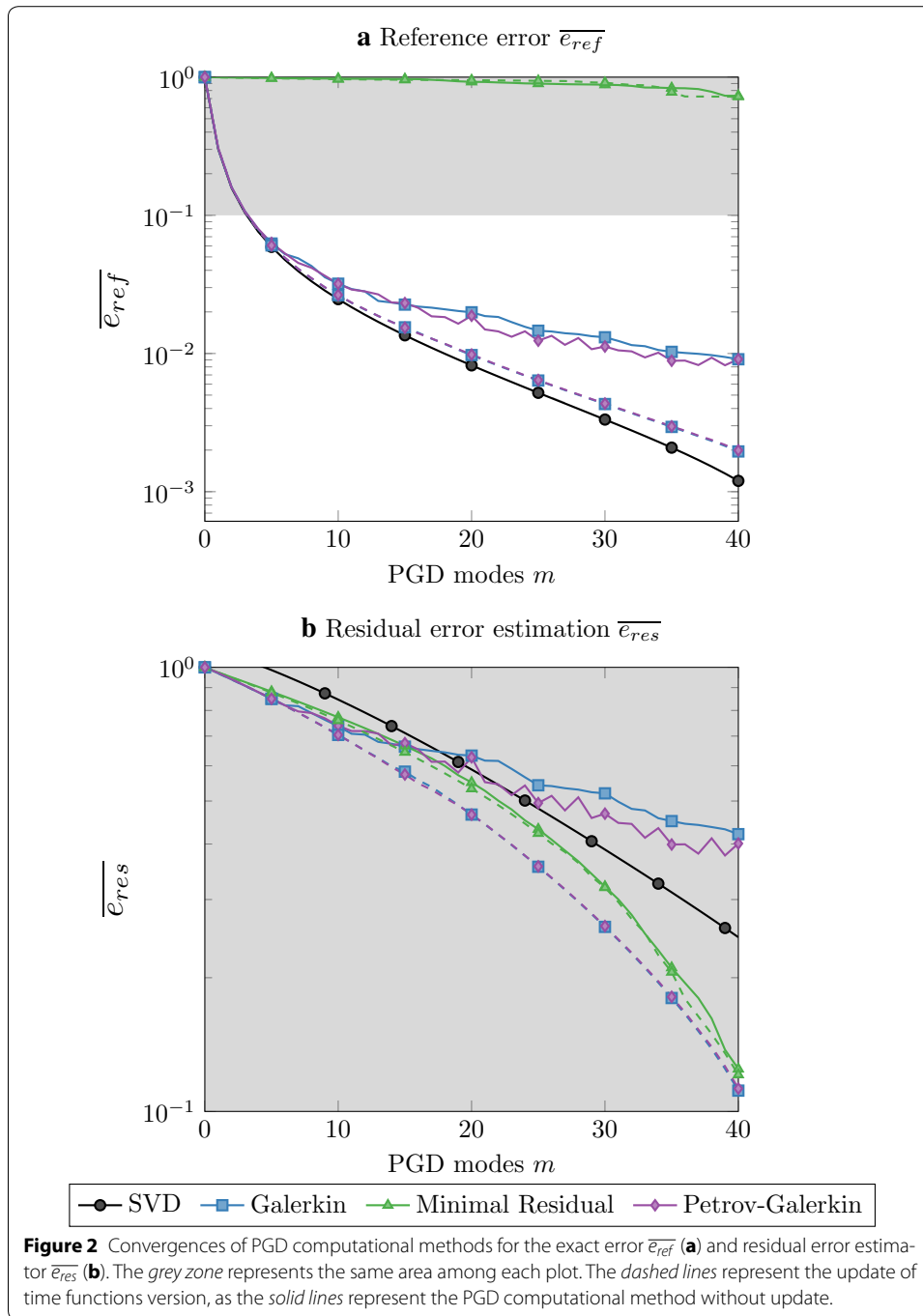
### Influence of the update phase

First we propose to compare the classical PGD computational methods presented in “Classical Proper Generalized Decomposition methods”, with or without the update step (see “Improvement of the decomposition”). Convergence curves are given in Figure 2 for a specific value of velocity set to  $v = L/2T$  for both reference and residual error estimators for classical PGD computational methods.

In Figure 2a, the first 10 modes of Galerkin/Petrov–Galerkin computational methods are close to the optimal decomposition obtained by SVD. Although extra modes degrade that convergence, they tend to follow the same path. Very poor convergence is observed for the classical Minimal Residual PGD computational method according to the reference error  $\overline{e_{ref}}$ . For Galerkin/Petrov–Galerkin PGD computational methods, the update step improves significantly the quality of the PGD approximation without fitting the optimal one from SVD. They seem to result in the same PGD decomposition as the error is identical. To get an error of 1%, those two methods need 20 modes when the SVD requires 18 modes.

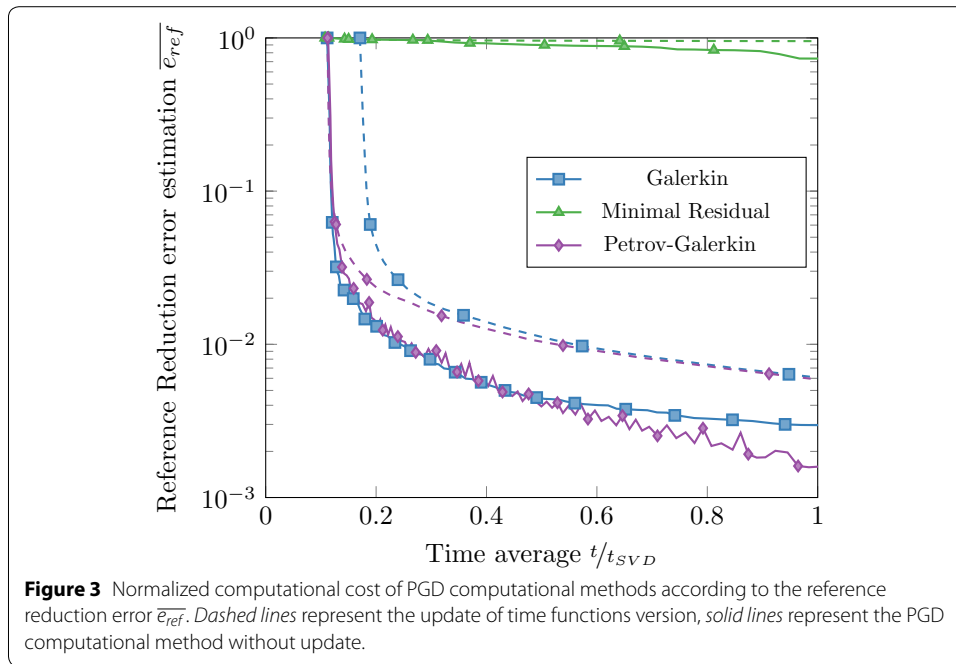
In general, the reference solution is unknown and convergence plots in Figure 2a cannot be obtained. Reduction error is therefore estimated through the residual error estimator  $\overline{e_{res}}$  as represented in Figure 2b. An estimation about 40 times larger than the reference one for all computational methods is observed. The residual error  $\overline{e_{res}}$  predicts that the best PGD decomposition comes from the Minimal Residual PGD computational method as it minimizes the residual, even if the reference reduction error says otherwise. The update step for this specific method does not improve the PGD decomposition. On the contrary, the update step highly improves Galerkin/Petrov Galerkin PGD computational methods to the same convergence curve, with a PGD decomposition better than the Minimal Residual PGD one. According to the residual error estimator, the SVD decomposition is not the optimal one, since we build it according to the energy norm rather than Euclidean norm.





As the purpose of reduction model is to reduce the computational cost of a brute force approximated solution, Figure 3 shows the computational cost of each PGD computational method where the time is averaged by  $t_{SVD}$ , the time required to compute a brute force solution and the separated form through SVD.

In Figure 3 the reference reduction error is plotted as a function of the averaged time to reach it. For the same error level, the consideration of the update step for Galerkin/ Petrov-Galerkin PGD computational method requires more computational time than



the classical method. However, when using the update step fewer PGD modes are required than without update, thus enhancing storage footprint of the solution and post-processing manipulation.

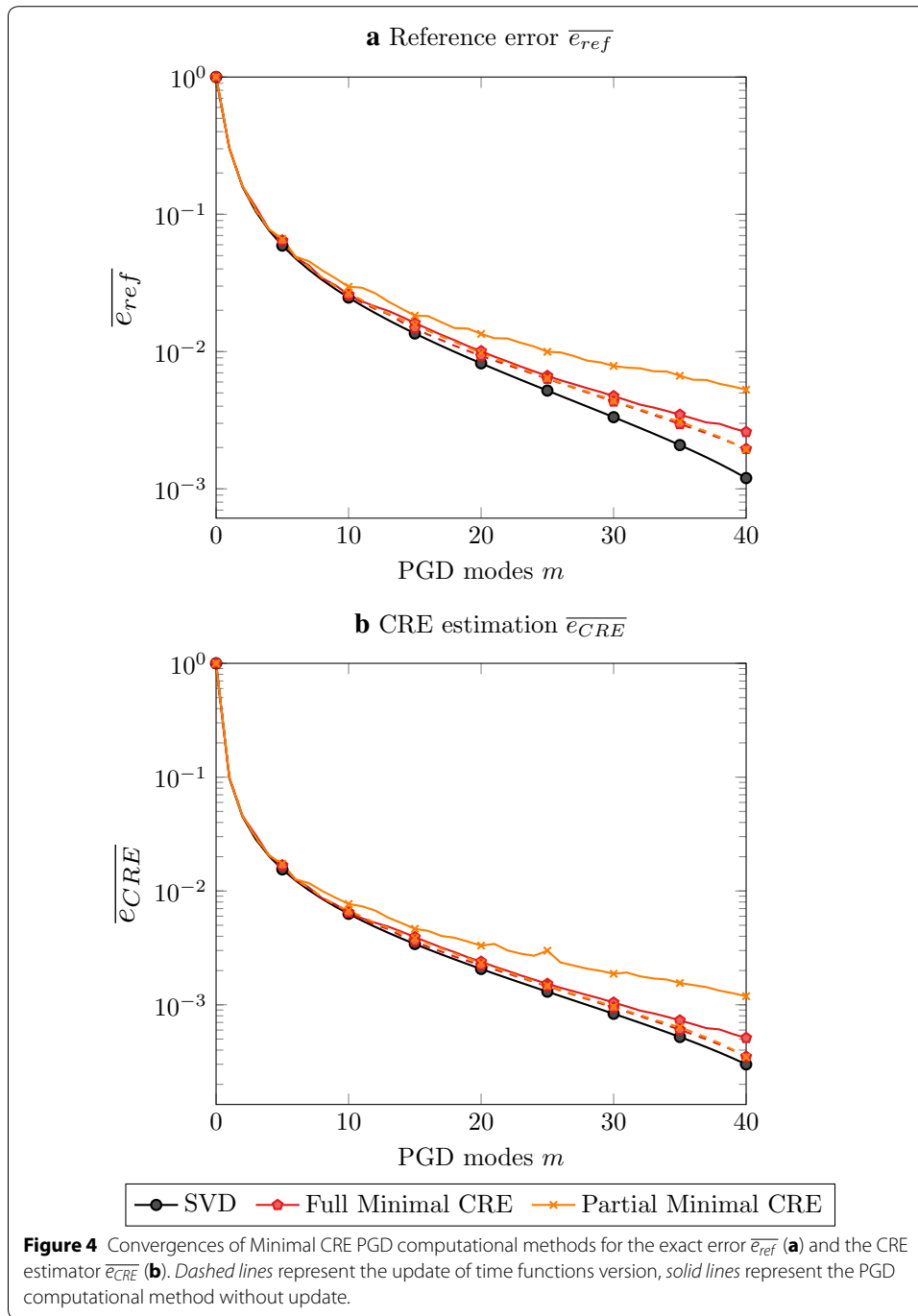
The update step improves drastically the PGD solution and, in the author opinion, should always be used. In the following, the PGD computational methods will only refer to the PGD one with time function update step, unless otherwise specified.

### Minimal CRE performances

In “[Minimisation of the Constitutive Relation Error](#)” we proposed a new PGD computational method based on the minimization of CRE, with two variants. In Figure 4, the convergence of each Minimal CRE PGD variant is represented with or without the update step.

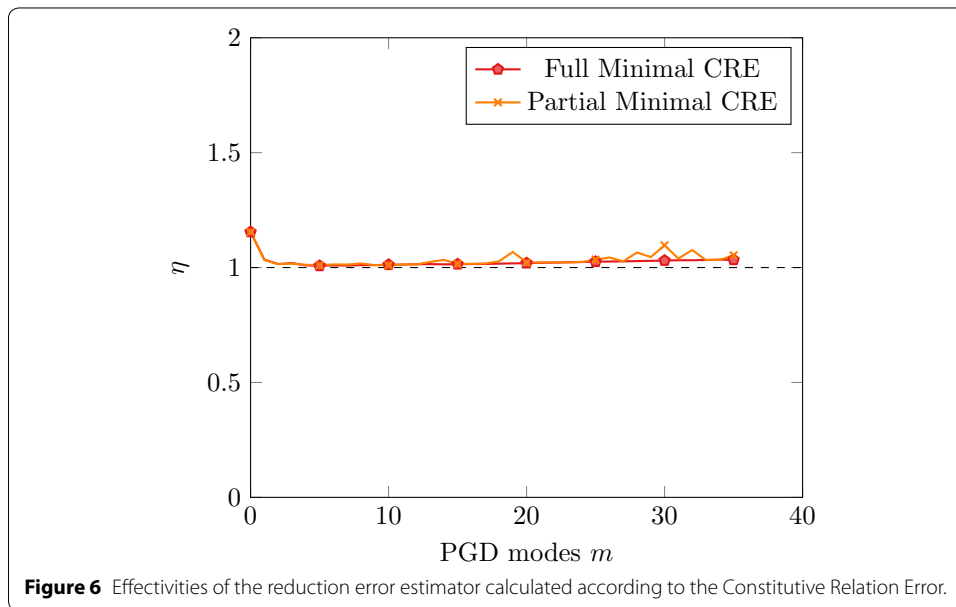
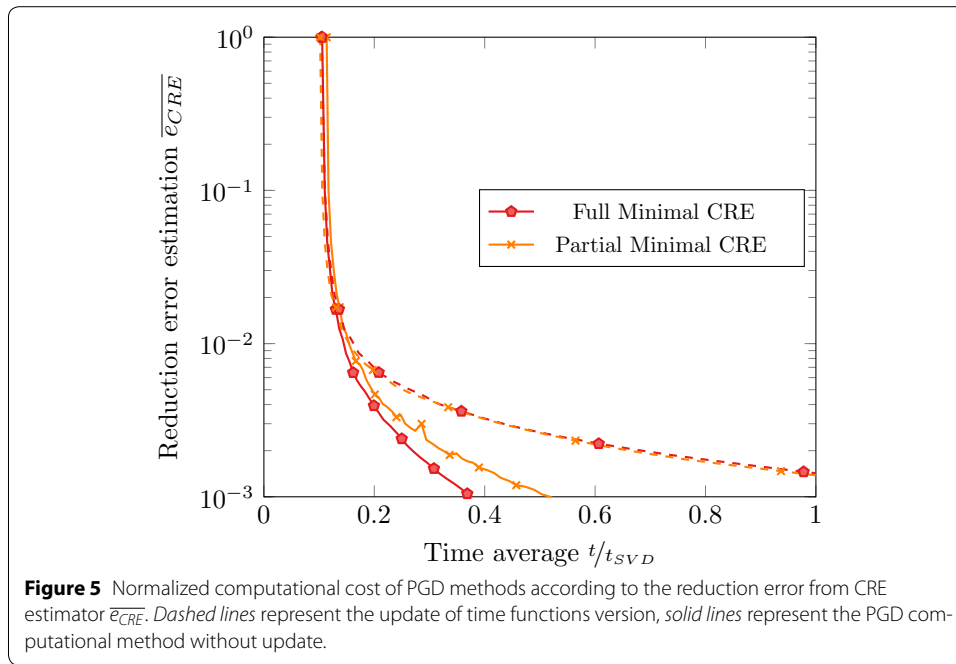
We observe in Figure 4a, that the first 10 modes of Full/Partial Minimal CRE PGD computational methods are close to the optimal decomposition obtained by SVD. However extra modes degrade that convergence as the full minimization is better than the partial one. One should observe that when the update step is used, full and partial minimizations have the same convergence rate. As a matter of fact, the partial minimization with update gives similar results as the full minimization with update, for a lower complexity (Figure 5). Despite the absence of computational time gain in 1D, we hope that the partial minimization will present a better computational time in higher dimension.

In general, the reference solution is unknown and convergence plots in Figure 4a cannot be obtained. Since the new proposed Minimal CRE PGD computational methods have their own reduction error estimation (see “[Minimisation of the Constitutive Relation Error](#)”) evaluated through the CRE indicator, the reduction error is therefore estimated through the CRE estimator  $\bar{e}_{CRE}$  as represented in Figure 4b. As seen in Figure 4a, the update step improves the convergence of the computational methods close to the



reference one. As a matter of fact, they are effective (accurate and guaranteed) as shown in Figure 6.

As a result, in the general case, one can only evaluate some error estimators depending on the computational methods. Then the CRE indicator obtained by minimizing the CRE leads to an improved and accurate reduction error estimation. Other PGD computational methods errors underestimate the reduction error. Unlike the Minimal CRE PGD solution, they require to compute extra modes to target a specific reference error.



The two CRE minimization strategies show great performances, robustness from minimization and give direct effective error estimator.

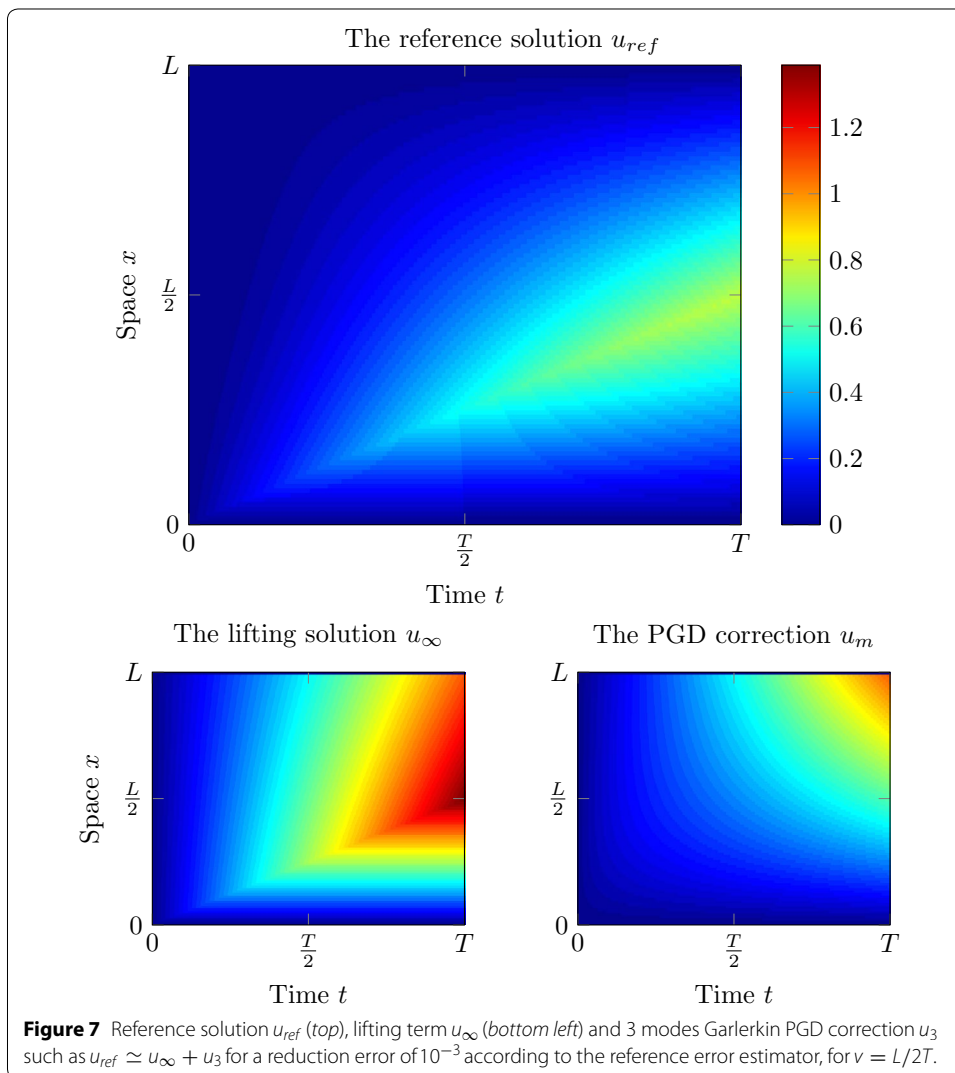
#### Limits of the PGD

In “[A non-separable lifting algorithm](#)”, we proposed a PGD computational method lifted with a non-separable known solution. Figure 7 represents the enrichment solution  $u_\infty$  solution over an infinite domain and the correction PGD solution  $u_m$  computed with a Galerkin PGD strategy with time functions update step. 3 modes are enough as the PGD

solution matches the reference solution  $u_{ref}$  below a relative reference error of  $10^{-3}$ . It is easy to observe that the correction term verifies the separation of variables from Figure 7, as the solution is smooth and continuous.

In Figure 8, the number of PGD modes required to obtain a reference error  $\overline{e}_{ref}$  below  $10^{-2}$  is studied depending on the loading velocity. While SVD and PGD computational methods require much more modes to catch the influence of the speed when the velocity increases, the lifting method only requires 2 PGD modes to get an accurate solution below a relative reference error of  $10^{-2}$ . Such properties come from the lifting solution  $u_\infty$  that catches the influence of the speed. According to (6), the 35th singular value of  $u_\infty$  is below  $10^{-2}$  for  $v = \frac{L}{2T}$ . It should be noted that when the velocity of the loading is greater than one, the flame leaves the domain. As a results, a plateau appears after that value.

Therefore, we rather propose to study the influence of the instationary term in the equilibrium equation (2b). To do so, we study the number of PGD modes required to meet a target error of  $10^{-2}$  according to the diffusion coefficient  $c/\mu$ , as shown in



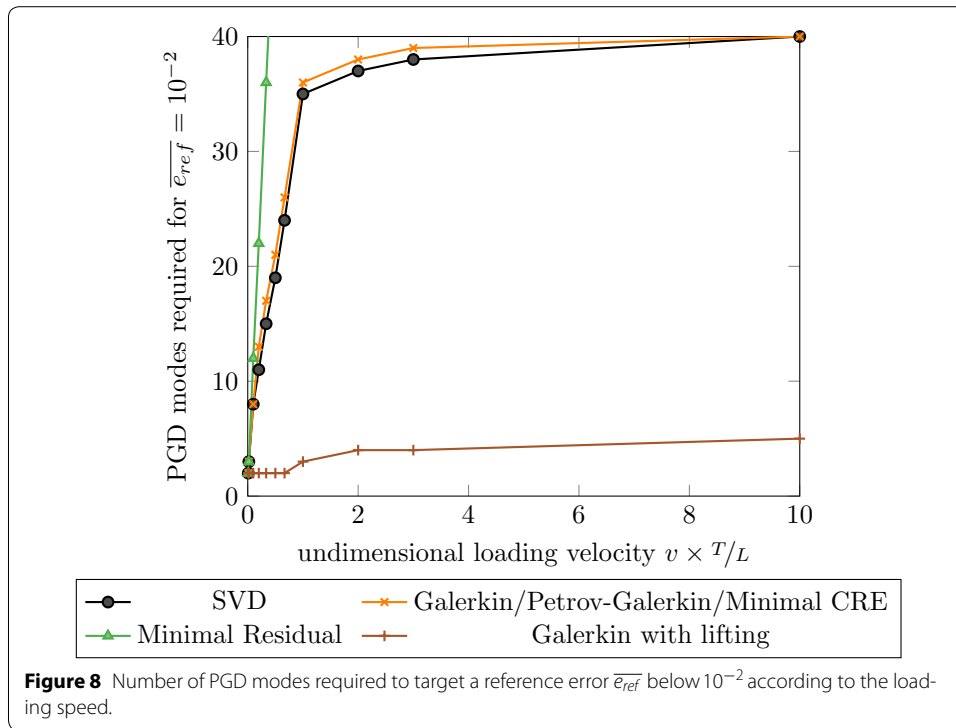


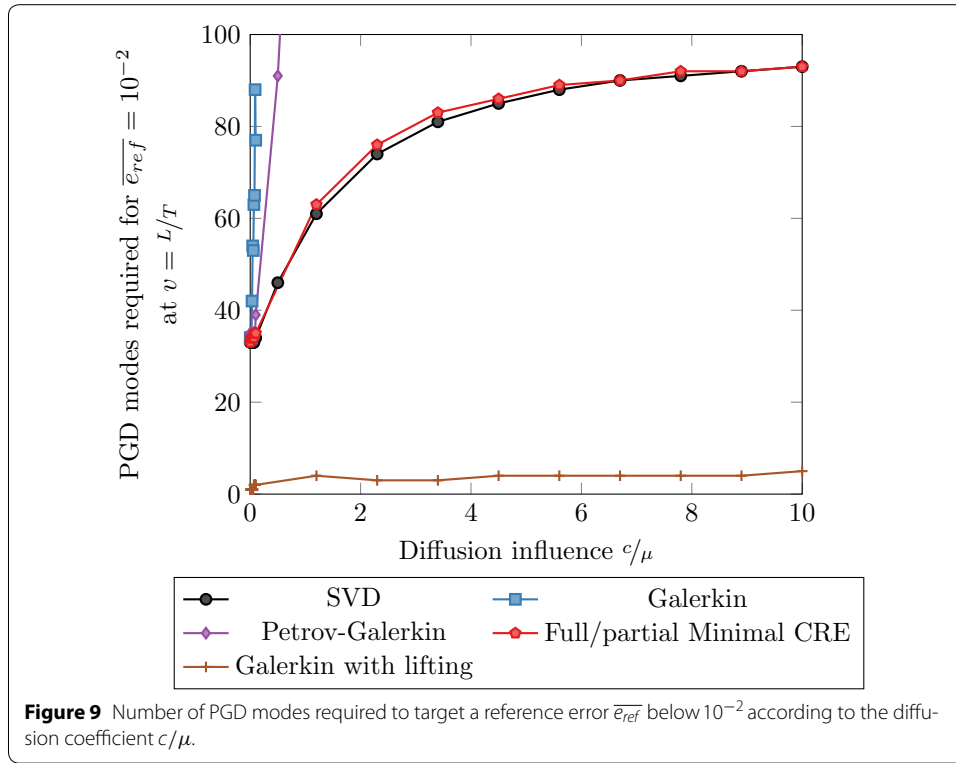
Figure 9. While classical PGD methods have difficulties to catch the right solution, minimizing the CRE leads to a better solution close to the optimal SVD one. As previously noted, the lifting technique requires much less modes to catch the solution, and does not depend of the influence of the diffusion term. Even if SVD or PGD computational methods require less modes to reach such an error, the lifting technique has the advantage not to compute those modes and therefore does not depend on the loading velocity.

From Figures 8 and 9, we study the influence of the transient part of the solution. Such a part does not verify the assumption of separated variables representation. When the diffusion term is neglectable, time does not influence the solution and a separated representation is implicit. On the other hand, time cannot be longer neglected, and extra separated modes are required.

In this section, the assumption of variable separation shows its limits for instationary problems that do not verify such an assumption. Initializing PGD strategies with a known unseparable solution circumvents that limit. Such more thoughtful strategy, presented as a lifting PGD technique, shows that the PGD method is not faulted.

## Conclusion

We reviewed different possible PGD computational methods for the a priori construction of reduced order time-dependent thermal model (ROM) based on the separated representation of the related solution. Different constructions have been analyzed, based on Galerkin, Minimal Residual and Petrov–Galerkin formulations



of the evolution problem. They were compared on a one dimensional benchmark problem.

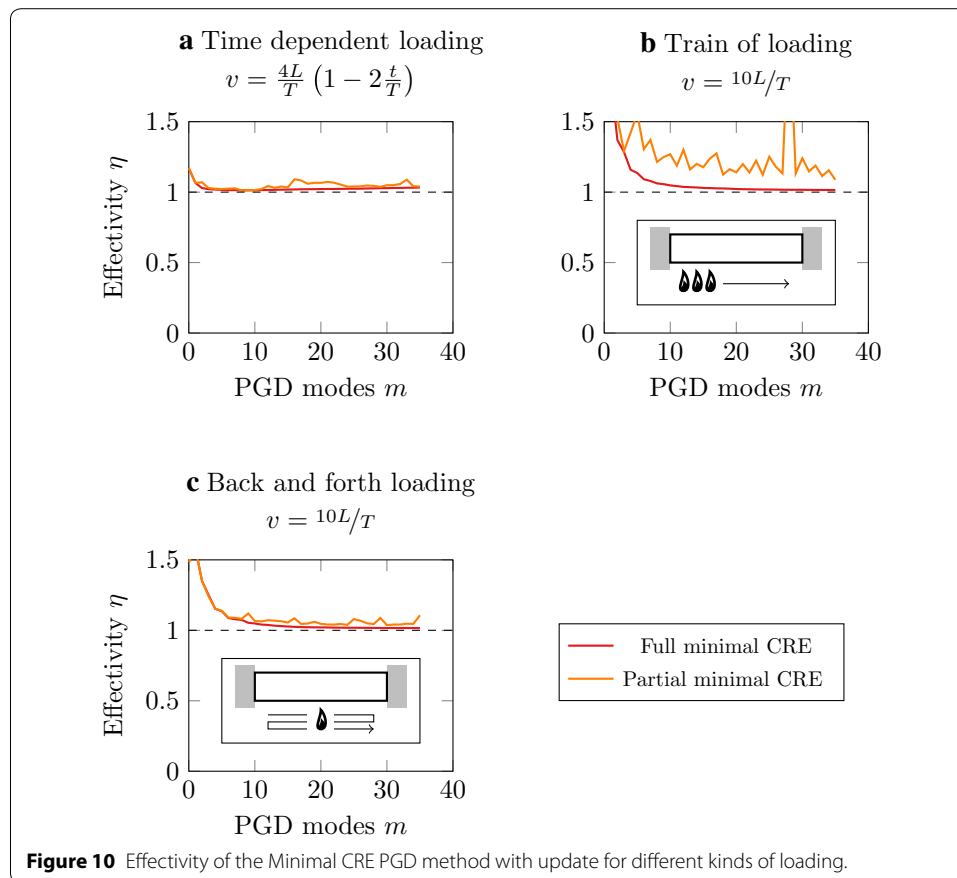
The numerical example has illustrated the importance of selecting a relevant and accurate reduction error estimator, as it is required to know when to stop the progressive algorithm. As the reference error estimator is generally out of reach, classical PGD computational methods usually lie on residual error estimator. As a matter of fact, this estimator is not effective nor accurate.

We proposed an innovative PGD construction based on minimizing the CRE. Its main advantage is to provide a robust, accurate and relevant reduction error indicator while maintaining advantages of PGD. Due to the use of CRE, discretization error is within easy reach and allows adaptive construction of PGD decomposition.

The update step highly improves convergence properties of all PGD computational methods. Putting aside the Minimal Residual PGD, all PGD computational methods coupled with this update step converged to the same PGD decomposition. It has the same convergence properties as an optimal reduced solution obtained by SVD.

Finally, limitations of PGD computational methods have been illustrated by lifting Galerkin PGD computational method with a non-separable solution. Convergence of PGD computational methods tends to decrease when the separation of variables assumption cannot respect the problem to solve. We illustrated it for a time dependent problem, but parameters dependent problems may have the same behavior.

Further analysis of the Minimal CRE PGD computational method was conducted for different loading types and confirms its performances as shown in Figure 10. Future



work will be devoted to the extension of the newly proposed PGD computational method (CRE) to non-linear and multi-parameter problems.

### Endnote

<sup>a</sup>Here,  $\mathbb{V}^\dagger$  denotes the adjoint matrix of  $\mathbb{V}$ .

### Authors' contributions

All authors have read and approved the final manuscript.

### Compliance with ethical guidelines

### Competing interests

The authors declare that they have no competing interests.

Received: 30 January 2015 Accepted: 29 June 2015

Published online: 16 July 2015

### References

1. Krysl P, Lall S, Marsden J (2001) Dimensional model reduction in non-linear finite element dynamics of solids and structures. *Int J Numer Methods Eng* 51(4):479–504
2. Ryckelynck D (2005) A priori hyperreduction method: an adaptive approach. *J Comput Phys* 202(1):346–366
3. Boyaval S, Bris CL, Maday Y, Nguyen NC, Patera AT (2009) A reduced basis approach for variational problems with stochastic parameters: application to heat conduction with variable robin coefficient. *Comput Methods Appl Mech Eng* 198(41):3187–3206



4. Kunisch K, Volkwein S (2001) Galerkin proper orthogonal decomposition methods for parabolic problems. *Numer Math* 90(1):117–148
5. Sanghi S, Hasan N (2011) Proper orthogonal decomposition and its applications. *Asia Pac J Chem Eng* 6(1):120–128
6. Maday Y, Rønquist E (2002) A reduced-basis element method. *J Sci Comput* 17(1–4):447–459. doi:10.1023/A:1015197908587
7. Rozza G, Huynh DBP, Patera AT (2008) Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Arch Comput Methods Eng* 15(3):229–275. doi:10.1007/s11831-008-9019-9
8. Chinesta F, Ladevèze P, Cueto E (2011) A short review on model order reduction based on proper generalized decomposition. *Arch Comput Methods Eng* 18(4):395–404. doi:10.1007/s11831-011-9064-7
9. Chinesta F, Ladevèze P (eds) (2014) Separated representations and PGD-based model reduction, vol CISM554. Springer, Udine
10. Ladevèze P (1989) The large time increment method for the analysis of structures with non-linear behavior described by internal variables. *Comptes Rendus de l'Académie des Sciences Série II* 309(11):1095–1099
11. Ladevèze P (1999) Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation. Springer, Berlin
12. Néron D, Ladevèze P (2010) Proper generalized decomposition for multiscale and multiphysics problems. *Arch Comput Methods Eng* 17(4):351–372
13. Ladevèze P (2014) PGD in linear and nonlinear computational solid mechanics. In: Chinesta F, Ladevèze P (eds) Separated representations and PGD-based model reduction, vol CISM554. Springer, Udine, pp 91–152
14. Nouy A (2008) Generalized spectral decomposition method for solving stochastic finite element equations: invariant subspace problem and dedicated algorithms. *Comput Methods Appl Mech Eng* 197(51):4718–4736
15. Niroomandi S, Alfaro I, Cueto E, Chinesta F (2012) Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models. *Comput Methods Programs Biomed* 105(1):1–12
16. Leygue A, Verron E (2010) A first step towards the use of proper general decomposition method for structural optimization. *Arch Comput Methods Eng* 17(4):465–472
17. Chinesta F, Ammar A, Cueto E (2010) Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Arch Comput Methods Eng* 17(4):327–350
18. Vitse M, Néron D, Boucard P-A (2014) Virtual charts of solutions for parametrized nonlinear equations. *Comput Mech* 54(6):1529–1539
19. Ladevèze P, Chamoin L (2011) On the verification of model reduction methods based on the proper generalized decomposition. *Comput Methods Appl Mech Eng* 200(23):2032–2047
20. Temlyakov VN (2008) Greedy approximation. *Acta Numer* 17:235–409
21. Ammar A, Chinesta F, Falcó A (2010) On the convergence of a greedy rank-one update algorithm for a class of linear systems. *Arch Comput Methods Eng* 17(4):473–486
22. Nouy A (2010) A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Comput Methods Appl Mech Eng* 199(23):1603–1626
23. Ladevèze P, Chamoin L (2013) Toward guaranteed PGD-reduced models. *Bytes and Science. CIMNE: Barcelona*, pp 143–154
24. Golub GH, Van Loan CF (2012) Matrix computations, vol 3. JHU Press, Baltimore
25. Hubert L, Meulman J, Heiser W (2000) Two purposes for matrix factorization: a historical appraisal. *SIAM Rev* 42(1):68–82
26. Bussy P, Rougée P, Vauchez P (1990) The large time increment method for numerical simulation of metal forming processes. In: *Proceedings of NUMETA*, pp 102–109
27. Chamoin L, Ladevèze P (2012) Robust control of PGD-based numerical simulations. *Eur J Comput Mech* 21(3–6):195–207
28. Allier P-E, Chamoin L, Ladevèze P, Pled F (2015) A posteriori error estimation and adaptive strategy for PGD reduced-order models applied to parametrized linear elliptic and parabolic problems (under review)

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)

---