Real time parameter identification and solution reconstruction from experimental data using the Proper Generalized Decomposition

E. Nadal[1], F. Chinesta[1], P. Díez[2,4], F.J. Fuenmayor[3] and F.D. Denia[3]

[1] Institut de Recherche en Génie Civil et Mecanique (GeM-UMR-CNRS)
ESI Group International Chair
École Centrale de Nantes (ECN)
1, rue de la Noë, 44300, Nantes, France
{Enrique.Nadal;Francisco.Chinesta}@ec-nantes.fr

[2] Laboratori de Càlcul Numèric (LaCàN)
Universitat Politècnica de Catalunya (UPC)
c/ Jordi Girona, 1-3, 08034, Barcelona, Spain
pedro.diez@upc.edu

[3]Centro de Investigación en Ingeniería Mecánica (CIMM)
Universitat Politècnica de València (UPV)
c/ Camino de Vera, s/n, 46022, Valencia, Spain
{ffuenmay,fdenia}@mcm.upv.es

[4] International Center for Numerical Methods in Engineering (CIMNE)
Campus Nord UPC, E-08034 Barcelona, Spain

Abstract

Some industrial processes are modelled by parametric partial differential equations. Integrating computational modelling and data assimilation into the control process requires obtaining a solution of the numerical model at the characteristic frequency of the process (real-time). This paper introduces a computational strategy allowing to efficiently exploit measurements of those industrial processes, providing the solution of the model at the required frequency. This is particularly interesting in the framework of control algorithms that rely on a model involving a set of parameters. For instance, the curing process of a composite material is modelled as a thermo-mechanical problem whose corresponding parameters describe the thermal and mechanical

1

behaviours. In this context, the information available (measurements) is used to update the parameters of the model and to produce new values of the control variables (data assimilation). The methodology presented here is devised to ensure the possibility of having a response in real-time of the problem and therefore the capability of integrating it in the control scheme. The Proper Generalized Decomposition is used to describe the solution in the multi-parametric space. The real-time data assimilation requires a further simplification of the solution representation that better fits the data (reconstructed solution) and it provides an implicit parameter identification. Moreover, the analysis of the assimilated data sensibility with respect to the points where the measurements are taken suggests a criterion to locate of the sensors.

1   Introduction

Any computational model is characterized by a set of input parameters. Once specific values are assigned to these parameters, the model is ready to be solved and it can be used to predict the behaviour of a system and also to select the corresponding input in order to produce the desired output (control problem). Numerical models for control must be capable of producing online responses required to the system. The online response aims at providing the numerical solution in a time delay that is much smallerthan the characteristic time of the phenomena described in the simulation. This computational time constraint is referred as real-time solution in the remainder of the paper. Thus, the modelling of systems for predictive control is much more demanding than for predictive design.

Moreover, in this context there is a real need of complementing the mathematical models by incorporating information from empirical data. This idea is frequently used in weather forecast simulations [1, 2, 3] and referred as data assimilation techniques. For complex models with a large number of material parameters involved, this leads to solving multi-parametric optimization problems. These simulations of practical interest tends to be computationally prohibitive for a standard model in computational mechanics. Different alternative approaches based on Model Order Reduction (MOR) techniques with basis constructed with the Proper Orthogonal Decomposition (POD) [4, 5, 1, 2] or the Empirical Orthogonal Functions (EOF) [6, 7] have been proposed in this framework for different physics as alternative to classical

methods. These techniques require an offline sampling of the parametric space, and appropriate enrichment procedures in order to ensure enough precision. When addressing a nonlinear problem, techniques of this kind are useful when the real time constraint is not a requirement. In the same context, Herzog and Riedel [8] propose a greedy algorithm for the sensor placement using the information of a POD solution with the purpose of reconstructing the thermo-mecanical model in real time when all parameters are a priori known. Other similar approaches, based on the POD, can be found in [9, 10] and references therein for different problems. The basic idea of these methods is, first, to obtain a reduced basis via POD, use it for the sensors placement and, then to reconstruct the solution online making use of the sensor measures. Chardon *et. al.* [11] proposed a methodology for the Helmholtz problem that projects the data taken from a set of sensors to a basis that is known *a priori*, constructed with Fourier-Bessel functions or plane waves. In this case the projection is carried out with a least squares based approach. Grepl *et. al* [12] introduced a methodology, based on the Empirical Interpolation Method (EIM) [13] for the real time reconstruction for nonaffine and nonlinear problems equipped with an *a posteriori* error estimation in order to control the quality of the reduced approximation. More recently, Maday *et. al.* [14] proposed a methodology, making use of the Generalized Empirical Interpolation Method (GEIM) [15]. This approach, namely Parametrized-Background Data-Weak (PBDW), takes information from both, the mathematical model and also from empirical data and it is able to work under real time constraint for a certain number of unknown parameters.

The Proper Generalized Decomposition (PGD) [16] is an alternative technique to build a basis for MOR that allows a real-time reconstruction of the solution and easily deals with multi-parametric models for industrial processes. When comparing PGD and POD, the main difference is that PGD is constructing an *a priori* reduced basis while solving the problem. On the contrary, the POD constructs an *a posteriori* reduced basis, taking the information from a set of solutions of the problem. In the case of the POD, the construction of the set of solutions is potentially problematic in the case of high dimensional spaces. The interested reader is addressed to [17] for a deeper analysis of this aspect. The PGD provides a multi-parametric solution [18, 19], explicitly depending on the parameters to be identified and avoids the *curse of dimensionality* issue when various parameters are considered. Accordingly, the PGD reduced order representation of the mathematical model under consideration produces not only a real-time solution, but also the identification of the material parameters. Additionally, it is accompanied by an error assessment tool, indicating the quality of the reconstructed

numerical solution.

The methods proposed in the present paper are based on the multi-parametric PGD solution of the mathematical problem, which is *a priori* evaluated offline. First, a reduced basis is extracted from this solution, which is used to obtain online real-time solutions for fast and multiple simulation queries. Secondly, a methodology is devised to find some smart sensor locations making use of the PGD-based reduced basis. This methodology is based on the EIM [13] an its discrete counterpart, the DEIM (Discrete Empirical Interpolation Method) [20] adapted to the PGD [19]. Thus, the third step consists of using online the PGD-based reduced basis to solve a small-size parameter identification problem fitting experimental data, and providing the estimated parameters and the reconstructed solution. This is accompanied by an error estimate, which can be computed before the data assimilation process, used to assess the number of terms in the selected reduced basis, and an error indicator to control the accuracy of the solution obtained.

The remaining of the paper is structured as follows: section 2 states the problems under consideration: (i) the boundary value problem, (ii) the solution reconstruction and parameter identification, and (iii) the sensor placement. Section 3 provides the details of the PGD algorithm. Section 4 describes how to obtain the reduced basis, the reconstructed solution, and the parameter identification procedure. Section 5 discusses the algorithm used to place the sensors in smart positions. Section 6 is devoted to the analysis of the proposed error indicators. Section 7 presents and discusses the numerical results. Finally section 8 draws the main conclusions and perspectives.

2 Mathematical model and problem statement

Here, a methodology is proposed to obtain a real-time solution for a certain boundary value problem (BVP) and to integrate the experimental data provided by a set of sensors. An algorithm to select the location of the sensors is also derived. Thus, this section discusses both the model, which is the convection diffusion equation, and the two specific problems under consideration:

- data assimilation

- sensor placement.

This section describes the problems and presents the notation and the following sections detail the methodology proposed to solve them.

2.1 Boundary value problem

The 2D quasi-static parametric convection-diffusion equation defines a boundary value problem which is considered to model the thermal behaviour of the system. The temperature field u is the primal unknown, the flux field is denoted by $\mathbf{q}$, and $\mathbf{v}$ is the velocity field, considered as input data. These fields take values in the domain $\Omega_x \times \Omega_y = \Omega \subset \mathbb{R}^2$, of boundary $\partial\Omega$ (the fact of considering a separable domain $\Omega = \Omega_x \times \Omega_y$ is not essential in that follows). Prescribed fluxes $\bar{\mathbf{q}}$ are imposed on $\Gamma_N \subset \partial\Omega$, while temperatures $\bar{u}$ are set on the complementary part $\Gamma_D \subset \partial\Omega$. Function f denotes the source term.

Thus, the unknown function $u(x, y)$ is such that

$$\mathbf{v} \cdot \nabla u + \nabla \cdot \mathbf{q} - f = 0 \qquad \text{in } \Omega \qquad (1a)$$

$$\mathbf{q} = -k\nabla u \qquad \text{in } \Omega \qquad (1b)$$

$$\mathbf{q} \cdot \mathbf{n} = \bar{q} \qquad \text{on } \Gamma_N \qquad (1c)$$

$$u = \bar{u} \qquad \text{on } \Gamma_D \qquad (1d)$$

where k, the diffusion coefficient, is considered to be an unknown parameter uniformly distributed in space.

For each value of the parameter k, problem (1) is equivalent to the following weak form: find $u \in (V + \{w\})$ such that

$$a(u, v) = l(v), \ \forall v \in V$$

$$\text{where} \qquad a(u, v) := \int_\Omega v(\mathbf{v} \cdot \nabla u) + \nabla v \cdot k\nabla u \ \mathrm{d}\Omega \qquad (2)$$

$$\text{and} \qquad l(v) := \int_\Omega v \ f \ \mathrm{d}\Omega + \int_{\Gamma_N} v \ \bar{q} \ \mathrm{d}\Gamma$$

where $V = \{v \ : \ v \in H^1(\Omega), v|_{\Gamma_D} = 0\}$ and w is a particular temperature field satisfying the Dirichlet boundary conditions (1d). Note that in this case the coefficient k has been taken as parameter. Actually, the time dependence is accounted for in the model through the variation of f (and k).The model is quasi-static in the sense that time does not appear explicitly in the problem statement. However, loads and the material properties evolve in time and hence the solution. This is equivalent to assume that the time scale for thermal diffusion is much shorter than the characteristic time of the thermal loads.

The aim of this contribution is to integrate computational modelling and data assimilation into the control process. Thus, the real-time parameter identification is an essential ingredient for the control strategy. Other parameters can also be included in the model, such as the intensity of the

source term or the advection term (velocity field), but for sake of simplicity they are not considered in the formulation considered in the present work.

The use of the standard FE framework for resolving the dependence on k would require solving a problem for each value k. In order to circumvent this issue, this work proposes the use of the PGD solver that is able to provide a parametric solution depending on the diffusion coefficient in one shot. The PGD solver for this problem is detailed in section 3.

2.2   Data assimilation

The interest is to obtain a solution to the problem (2) under real time constraint and integrating the data provided by a set of sensors. Thus, the data assimilation problem reads as follows: given a set of n_{M} different measures provided by the sensors:

$$u_{\mathrm{M}}(\mathbf{s}_i) \text{ for } i = 1, \dots, n_{\mathrm{M}} \tag{3}$$

placed at points $\mathbf{s}_i = (x_i, y_i)$, the objective is to obtain an assimilated solution u_{A} such that:

$$u_{\mathrm{A}}(\mathbf{s}_i) \approx u_{\mathrm{M}}(\mathbf{s}_i) \text{ for } i = 1, \dots, n_{\mathrm{M}} \tag{4}$$

and also an estimation for the unknown parameter, in this case k, best fitting the measures. The process to obtain the assimilated solution is described in section 4. The accuracy and robustness of the assimilated solution is sensitive not only to the number of measures n_{M} but also to the placement of the sensors $\mathbf{s}_i$. Accordingly, the problem of finding the sensor placement is stated in next section, minimising the sensitivity of the assimilated solution to the sensor position.

2.3   Sensor placement

The input of the problem introduced in section 2.2 consists of experimental data provided by a set of sensors. In the case the experiment can be designed by setting the location of the measurement points (where to locate the sensors), the question of identifying the sensor location in order to minimize the influence of the measurement error should be addressed. The proposed methodology, described later, provides a reconstructed solution which is obviously affected by the measurement errors contained in the input data.

Thus, the objective of the sensor placement problem is to determine the locations $\mathbf{s}_1, \dots, \mathbf{s}_{n_M}$, of the n_{M} sensors minimising the sensitivity of the recovered solution with respect to the measurement errors.

Moreover, in the case the algorithm used to place the sensors is based on a greedy strategy, see for instance [8], the first sensors may be located independently of the new ones to be added later. That is, new sensors do not modify the position of the previous ones. Strictly speaking, this last property is not required but, in practice, it eases the process of adding new sensors and the implementation complexity of the algorithm. Section 5 contains a full description of this problem and the techniques used to solve it.

3   Proper Generalized Decomposition

3.1   Parametric solution

As indicated before, the solution of equation (2) must be evaluated for each value of the diffusion coefficient. This process leads to an unaffordable computational cost. The PGD technology applied to solve PDEs allows to solve the convection-diffusion problem, for any diffusion coefficient, in one shot. Following former works (see [18, 21, 22, 19] and the references therein) one could consider model parameters (as well as boundary conditions, initial conditions and geometrical parameters) as extra-coordinates, in order to calculate a multi-parametric solution. Thus, in order to compute the solution of problem (2) for any value of the diffusion coefficient k, it suffices considering k as an extra-coordinate (like x or y) and looking for $u(x, y, k)$ as described in [17]. The parametric solution reads:

$$u_{\text{PGD}}^{\hat{n}}(x, y, k) = \sum_{i=1}^{\hat{n}} \hat{X}_i(x) \hat{Y}_i(y) \hat{K}_i(k) \tag{5}$$

where $\hat{n}$ is the number of modes of the approximated solution. The higher the number of modes considered, the higher the quality of the solution will be, but with an increase of the computational effort. In this work it is considered that the number of modes used is sufficient to allow neglecting the difference between the Finite Element and PGD solutions. However, if required, different error estimators for the PGD solution are available in literature [23, 24, 25] in order to guarantee that the PGD solution is accurate enough. Note that the space domain has been represented as a tensor product. This is a general case in which all spatial dimensions have been separated. Space separation is of special interest for problems with distorted geometries as in the case of plates and shells [26, 27] or flows into laminates [28] in which the space separation (in-plane-out-of-plane) is highly recommended. In these situations the PGD is able to provide a 3D solution at the cost of 2D solutions. This increases the efficiency with respect to standard

techniques and avoids the use of highly distorted 3D meshes. In any case as previously indicated, the separated representation of space coordinates is not necessary for deriving the strategy proposed in this work.

In order to evaluate each mode a greedy algorithm is used. At enrichment step m of the PGD algorithm, the approximation $m-1$ is already computed:

$$u_{\mathrm{PGD}}^{m-1}(x, y, k) = \sum_{i=1}^{m-1} \hat{X}_i(x) \hat{Y}_i(y) \hat{K}_i(k) \tag{6}$$

and the new one is obtained, i.e

$$u_{\mathrm{PGD}}^{m}(x, y, k) = u_{\mathrm{PGD}}^{m-1}(x, y, k) + \hat{X}_m(x) \hat{Y}_m(y) \hat{K}_m(k). \tag{7}$$

An alternating direction iterative scheme is then used to solve the nonlinear problem for $\hat{X}_m(x)$, $\hat{Y}_m(y)$ and $\hat{K}_m(k)$. At iteration p, $\hat{X}_m^p(x)$ is computed from $\hat{Y}_m^{p-1}(y)$ and $\hat{K}_m^{p-1}(k)$, then $\hat{Y}_m^p(y)$ from $\hat{X}_m^p(x)$ and $\hat{K}_m^{p-1}(k)$, and finally $\hat{K}_m^p(k)$ from $\hat{X}_m^p(x)$ and $\hat{Y}_m^p(y)$. For further details of this method the interested reader is addressed to [17].

The corresponding strong form related to the problems involving the space functions results in a one-dimensional convection-diffusion-reaction equation with a source term, for which quasi-optimal stabilization methods exist [29] if needed. The interested reader can refer to [30] for a deeper analysis of this topic and numerical tests proving the performance of this approach.

3.2 PGD compact solution

When solving non-symmetric problems by using the separated representation constructor just described, the solution contains more modes than those provided by the Singular Value Decomposition (SVD) (or its multidimensional counterpart in the multi-dimensional case, the High Order Singular Value Decomposition (HOSVD)) applied to the problem solution computed by using a standard discretization technique. The decomposition provided by the PGD constructor is not optimal. This means that some modes do not have and important contribution to the solution reconstruction, thus being not necessary in the reduced approximation basis.

Therefore, a post-compression should be envisaged for expressing the solution in a more compact form. The post-compression is carried out by solving the following problem:

$$\int_{\Omega_x \times \Omega_y \times \Omega_k} v(u_{\mathrm{PGD}}^n - u_{\mathrm{PGD}}^{\hat{n}}) \ \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}k = 0, \quad \forall v \in L_2(\Omega_x) \times L_2(\Omega_y) \times L_2(\Omega_k). \tag{8}$$

With this aim, as described in chapter 3 of [17], the PGD is applied for approximation purposes. Since, in this case, the problem solved is symmetric, the separated solution is optimal in the sense that it approaches the one obtained by involving the SVD (when considering two coordinates) or its HOSVD counterpart when considering more than 2 coordinates. In the last case (more than two coordinates) there is not optimal decomposition, but our experience indicates that PGD and HOSVD leads to a similar number of modes in the decomposition. Thus, the solution of (8) can be used to define the reduced basis.

The solution $u_{\mathrm{PGD}}^n = \sum_i^n X_i(x) Y_i(y) K_i(k)$ approximates $u_{\mathrm{PGD}}^{\hat{n}}$ with $n \leq \hat{n}$ terms. The accuracy level of u_{PGD}^n, even with fewer terms, is similar to the one of $u_{\mathrm{PGD}}^{\hat{n}}$. This means that this compression algorithm gets a better basis for the solution. The approximation basis resulting of this post-compression process defines the reduced basis to be considered for identifying parameters and reconstructing solutions.

4 Data assimilation. Reconstruction of the solution

In this section, as indicated in section 2.2, the proposed algorithm to obtain a real-time reconstructed solution from the data provided by the sensors is introduced. It is supposed that the parametric PGD solution u_{PGD}^n is available. In order to obtain an assimilated solution it is possible to proceed in two different ways. The first one considers the full PGD solution and only adjusts the parameter k to the data provided by the sensors. The second option, the one used in this work, only uses a reduced part of the PGD solution as a basis to project the data provided by the sensors.

4.1 Option 1. Considering the full PGD basis

This first option assumes that the PGD solution perfectly describes the physical phenomena. This approach uses the full PGD solution and evaluates the parameter k which makes that the full PGD solution best fits to the measures in a least squares sense. That is:

$$u_{\mathrm{A}}(x, y) = u_{\mathrm{PGD}}^n(x, y, k_{\mathrm{A}}) \tag{9}$$

where k_{A} is obtained minimizing the following functional:

$$\mathcal{J}_0(k) = \sum_j^{n_{\mathrm{M}}} \left(u_{\mathrm{M}}(\mathbf{s}_j) - u_{\mathrm{PGD}}^n(x_j, y_j, k)\right)^2. \tag{10}$$

The minimization of the functional defined in (10) leads to a nonlinear equation to be solved in real time. Moreover, this methodology assumes that both the mathematical model and the PGD solution of the model accurately describe the physical phenomena. In some practical situations this assumption could be strong since the physical phenomena could not be perfectly modelled with the mathematical equation [14]. Additionally, in complex situations, the solution is extremely sensitive to the value of the parameter.

4.2 Option 2. Considering a subset of the PGD spatial basis

In order to increase the efficiency of the previous approach, a new reduced basis is used, consisting in the space modes of the PGD solution, ignoring the parametric function $K_i(k)$ for $i = 1, \dots, n$. The solution is sought as the linear combination of the elements of this basis better fitting the data provided by the sensors. This strategy is adopted in this work and is conceptually divided in two steps: 1) the construction of the reduced basis and 2) the characterization of the best fitted approximation to the data.

4.2.1 Reduced basis construction

The reduced basis of n_{R} terms $\mathcal{B}_{n_{\mathrm{R}}}$ is built with the first $n_{\mathrm{R}} < n$ space modes of the full PGD solution. That is, the procedure consists of:

1. Choosing the modes X_i and Y_i, $i = 1, 2, \dots, n_{\mathrm{R}}$, from the post-compressed PGD solution.

2. Defining the reduced basis, $\mathcal{B}_{n_{\mathrm{R}}}(x, y)$ as:

$$\mathcal{B}_{n_{\mathrm{R}}}(x, y) = [X_1(x)Y_1(y) \quad X_2(x)Y_2(y) \quad \dots \quad X_{n_{\mathrm{R}}}(x)Y_{n_{\mathrm{R}}}(y)]. \tag{11}$$

Once the reduced basis $\mathcal{B}_{n_{\mathrm{R}}}$ is built, the next step is to obtain the assimilated diffusion coefficient k_{A}, and also the linear combination of the functions involved in the reduced basis to build the assimilated solution: $u_{\mathrm{A}}(x, y) = \mathcal{B}_{n_{\mathrm{R}}}(\mathbf{x})\boldsymbol{\alpha}$, $\boldsymbol{\alpha} \in \mathbb{R}^{n_{\mathrm{R}}}$. The data assimilation process incorporates the information provided by the sensors, u_{M}, in order to determine k_{A} and $u_{\mathrm{A}}(x, y)$.

4.2.2 Data assimilation

When the reduced basis $\mathcal{B}_{n_R}$ is available, the online process that identifies, in real-time, the model parameter and reconstructs the solution starts by evaluating the coefficients of the reduced basis and also the model parameter from the data provided by the sensors. In order to get the coefficients for the reduced basis, a classical approach could be a least squares fitting to the data provided by the sensors, minimizing the following functional:

$$\mathcal{J}_1(\boldsymbol{\alpha}) = \sum_j^{n_M} \left(u_M(\mathbf{s}_j) - \sum_i^{n_R} X_i(x_j)Y_i(y_j)\alpha_i \right)^2. \tag{12}$$

The coefficients $\boldsymbol{\alpha}$ obtained fit as best as possible, in a least square sense, the data provided by the sensors. This approach ignores the mathematical model (1) because it only constitutes a simple data fitting. Since the PGD solution is available, (12) is modified by introducing the functions $K_i(k)$ in the minimization problem, according to:

$$\mathcal{J}_2(\boldsymbol{\epsilon}, k) = \sum_j^{n_M} \left(u_M(\mathbf{s}_j) - \sum_i^{n_R} X_i(x_j)Y_i(y_j)(K_i(k) + \epsilon_i) \right)^2. \tag{13}$$

In this last expression the coefficients α_i have been substituted by a new expression:

$$\alpha_i = K_i(k) + \epsilon_i \tag{14}$$

where k is the material parameter and ϵ_i takes into account the possible discrepancies with the model. This allows evaluating the assimilated material parameter k_A at the same time as $\boldsymbol{\epsilon}$. Compared with (10), this last functional provides more flexibility with the measurements, since the material parameter is not the only unknown.

Additionally, a weighted regularization term is added to the functional (13):

$$\mathcal{J}(\boldsymbol{\epsilon}, k) = \sum_j^{n_M} \left(u_M(\mathbf{s}_j) - \sum_i^{n_R} X_i(x_j)Y_i(y_j)(K_i(k) + \epsilon_i) \right)^2 + \tau \sum_i^{n_R} \epsilon_i^2. \tag{15}$$

The value of τ plays an important role and depends on the application. Large values of τ provide an assimilated solution closer to the solution provided by the PGD. On the contrary, small values of τ penalize the discrepancy

with the measures. Therefore, the choice of τ depends on the degree of confidence associated with the PGD solution or with the measures. Introducing $\tau > 0$ in the formulation, allows using $n_R > n_M$. This is not possible without the regularization term. In order to obtain the values of $\boldsymbol{\epsilon}$ and k_A the functional defined in (15) is made stationary, first with respect to $\boldsymbol{\epsilon}$, leading to a linear system of equations of size n_R, defined as follows:

$$\mathbf{A}\boldsymbol{\epsilon} = \mathbf{b} \tag{16}$$

with:

$$\mathbf{A} = \sum_{j}^{n_M} \mathcal{B}_R^T(\mathbf{s}_j)\mathcal{B}_R(\mathbf{s}_j) + \tau\mathbf{I}_R, \tag{17}$$

$\mathbf{I}_R$ being the identity matrix of size n_R, and

$$\mathbf{b} = \sum_{j}^{n_M} \left(\mathcal{B}_R^T(\mathbf{x}_j)u_M(\mathbf{s}_j) - \mathcal{B}_R^T(\mathbf{s}_j)\mathcal{B}_R(\mathbf{s}_j)\mathbf{K}(k) \right) \tag{18}$$

with $\mathbf{K}(k) = [K_1(k) \ K_2(k) \ ... \ K_R(k)]^T$ and secondly, with respect to k, leading to a scalar nonlinear equation:

$$\sum_{j}^{n_M} \left(\mathcal{B}_R(\mathbf{s}_j)\frac{\mathrm{d}\,\mathbf{K}(k)}{\mathrm{d}\,k} \right) (u_M(\mathbf{s}_j) - \mathcal{B}_R(\mathbf{s}_j)\left(\mathbf{K}(k) + \boldsymbol{\epsilon}\right)) = 0 \tag{19}$$

whose resolution can be carried out with any standard nonlinear solver. Thus, a linear system of equations of size n_R has to be solved complemented by a nonlinear scalar equation. This is solved with an iterative staggered fixed-point scheme.

5 Sensor placement

Most of the industrial processes require a control system in order to update the working parameters of the process. For instance, in an oven it is interesting to control the temperature at certain critical points, by adjusting the power of the heaters. During the construction of any industrial device such as an oven, the placement of the sensors controlling the process is crucial in order to get an accurate feedback of the process and to guarantee a proper monitoring. Sometimes physical or practical constraints may preclude placing the sensors in the desired locations. However, in the current investigation, no physical constraints have been considered. But, they can be easily introduced in the method by restricting the search space.

In this section the methodology used to evaluate the position for the sensors is detailed as indicated in section 2.3. It is assumed that systematic measurement error, $\delta = -u + u_\mathrm{M}$, is independent of the magnitude of the measure. As an illustrative example, we consider the case in which the solution could be expressed in one single PGD mode. Correspondingly, a single sensor is also taken into account, $u_\mathrm{M}(\mathbf{s}_1)$ at point $\mathbf{s}_1 = (x_1, y_1)$. The objective is to evaluate the material parameter k_A such that the error of measure has the minimum influence. For that purpose the mode of the PGD solution is used and since it represents the measure $u_\mathrm{M}(\mathbf{s}_1) - \delta$:

$$u_\mathrm{M}(\mathbf{s}_1) = X(x_1)Y(y_1)K(k_\mathrm{A}) + \delta \qquad (20)$$

from which the function $K(k_\mathrm{A})$ can be evaluated:

$$K(k_\mathrm{A}) = \frac{u_\mathrm{M}(\mathbf{s}_1) - \delta}{X(x_1)Y(y_1)}. \qquad (21)$$

As it can be deduced from (21), the larger is the value of $X(x_1)Y(y_1)$, the smaller influence the measure error has. Therefore, the most convenient location of the sensor corresponds to taking $\mathbf{s}_1$ maximizing $X(x_1)Y(y_1)$. The idea behind of the previous academic example is equivalent to use EIM and its discrete counterpart the DEIM: once the reduced basis has been defined, the EIM/DEIM places the sensors in the locations providing the maximum amount of information.

In the general case of placing n_M sensors, the first n_M modes of the solution are considered. The first sensor is situated where the first mode reaches its maximum:

$$(x_1, y_1) = \arg \max_{(x,y)} |X_1(x)Y_1(y)|. \qquad (22)$$

Once the first position is obtained, the second sensor is placed where the following residual is maximum:

$$(x_2, y_2) = \arg \max_{(x,y)} |R_2(x, y)| \qquad (23)$$

where $R_2(x, y) = X_2(x)Y_2(y) - d_{1,2}X_1(x)Y_1(y)$. The scalar $d_{1,2}$ is obtained as follows:

$$d_{1,2} = \frac{X_2(x_1)Y_2(y_1)}{X_1(x_1)Y_1(y_1)}. \qquad (24)$$

By construction, $R_2(x, y)$ maximizes the information that the first mode cannot describe. For the other sensors ($j = 3, 4, ..., n_\mathrm{M}$), provided the location

13

of the previous ones, the position of each new sensor is obtained in a similar way:

$$(x_j, y_j) = \arg \max_{(x,y)} |R_j(x,y)| \qquad (25)$$

where $R_j(x,y) = X_j(x)Y_j(y) - \sum_{i=1}^{j-1} d_{i,j} X_i(x)Y_i(y)$. The values $d_{i,j}$ are obtained, for each sensor j, by solving the following linear system:

$$R_j(x_k, y_k) = 0 \quad \forall k = 1, 2, ..., j-1. \qquad (26)$$

With this algorithm, the locations of the n_M sensors are sequentially obtained. This means that the proposed algorithm does not provide a global optimal location[1] for the sensors due to its high computational cost, but a restricted optimal in a greedy sense for the given basis, as other proposed approaches do [9, 8]. As shown in the following, the number of sensors is not crucial in the quality of the solution as in other techniques [8] if the model solution is accurately represented with the PGD solution.

6 Quantifying the accuracy of the reduced basis

In section 5 the EIM/DEIM has been adapted to the PGD environment to predict in advance the best placement of the n_M sensors. At this point the interest is assessing the size of the reduced basis n_R to reach some given level of accuracy. As a first approach, the proposal is to use the following *safe* upper bound error measure to predict the size of the reduced basis, which can be evaluated before starting the data assimilation process. Note that, because of the use of the regularization term in expression (15), n_R can be smaller, equal or larger than n_M and the minimization problem is in all cases well-posed.

$$\mathcal{E}_{UB}(n_R) = \max_k \left\| \sum_i^n X_i(x)Y_i(y)K_i(k) - \sum_i^{n_R} X_i(x)Y_i(y)K_i(k) \right\|_2. \qquad (27)$$

Expression (27) represents the difference between the full PGD solution and the solution with only the first n_R terms. Following this error measure, when aiming at a prescribed error η, the criterion to decide upon the size of

---

[1]In this context optimal locations refers to the locations where the error is decreased as much as possible or, equivalently, where the amount of information captured is maximized.

the reduced basis consists in enforcing

$$\mathcal{E}_{UB}(n_{\mathrm{R}}) < \eta. \tag{28}$$

As indicated before, the expression (27) corresponds to an error upper bound. This is because, it considers the maximum for any possible value of k. However it would be more interesting obtaining a relation between this upper bound and the assimilated solution u_{A}. Let us assume that the full PGD solution with n terms exactly represents the measures. The following expression defines the error for the solution provided by the reduced basis:

$$\mathcal{E}_{EX} = \left\| \sum_i^n X_i(x) Y_i(y) K_i(k_{\mathrm{ex}}) - u_{\mathrm{A}}(x,y) \right\|_2. \tag{29}$$

The previous expression accounts for the exact error of the reduced solution but, in practice, it is not possible to evaluate it because the exact parameter value k_{ex} is not known. A computable error indicator is readily introduced as:

$$\mathcal{E} = \left\| \sum_i^n X_i(x) Y_i(y) K_i(k_{\mathrm{A}}) - u_{\mathrm{A}}(x,y) \right\|_2 \tag{30}$$

where the material parameter obtained from the minimization problem, k_{A}, is used. As it is demonstrated in the following, the error committed in the evaluation of the parameters is much lower than the error due to the truncation of the PGD basis. Thus, the indicator $\mathcal{E}$ provides a proper error assessment. Additionally, in the particular case of $\tau = 0$, it is easily proven that:

$$\mathcal{E} \le \mathcal{E}_{UB}(n_{\mathrm{R}}). \tag{31}$$

The coefficients $\boldsymbol{\alpha}$ affecting the functions of the reduced basis are selected to minimize the error. Any other choice, such as $K_i(k_{\mathrm{A}})$, provides larger error. In the case where $\tau \neq 0$, the upper bound property, which holds in practice, cannot be strictly guaranteed.

Additionally, in order to check the behaviour of the proposed technique, the following error measure defined:

$$\mathcal{E}_{EX,FEM} = \left\| u_{\mathrm{FEM}}(x,y;k_{\mathrm{ex}}) - u_{\mathrm{A}}(x,y) \right\|_2. \tag{32}$$

This error measure indicates the total difference between the Finite Element solution for the exact value of the parameter, k_{ex}, which is taken as reference solution in the numerical results, and the assimilated solution, u_{A}.

15

7 Numerical results

In order to check the proposed method, several numerical tests on a convection-diffusion problem have been carried out in quasi-static conditions. The online use in transient models is a work in progress where parameter identification and control strategies must be carefully addressed. The space domain is $\Omega_x \times \Omega_y = (0, 2) \times (0, 1)$ and the domain associated with the diffusion coefficient is $\Omega_k = (0.01, 0.1)$. A constant velocity field $\mathbf{v} = (1, 1)^T$ has been also considered. The Dirichlet boundary conditions, that induce a boundary layer in the vicinity of the boundary, are:

$$
\begin{cases}
u(0, y) = 0 \ \forall y \in \Omega_y \\
u(x, 0) = 0 \ \forall x \in \Omega_x \\
u(2, y) = \dfrac{y}{2} \ \forall y \in \Omega_y \\
u(x, 1) = x \ \forall x \in \Omega_x
\end{cases}
\tag{33}
$$

and the source term:

$$
f(x, y, k) = 20 e^{-100(x-1)^2} e^{-100(y-0.5)^2}
\tag{34}
$$

Figure 1 shows the reference solution, obtained with the Finite Element Method with a mesh of 10^4 bilinear rectangular elements with a size of 0.02 in the x and 0.01 in the y direction, for a value of the diffusion coefficient $k_{\mathrm{ex}} = 0.037$. From now on, this solution will be taken as reference. Moreover, the measures will be taken from this solution. Note that in practice the measures provided by the sensors will be taken from the physical component.

In what follows, the different bricks constituting the proposed approach are addressed. First, the sensor placements for a different number of sensors is defined. Next, the reconstruction of the solution for different number of sensors and also for different size of the reduced basis is considered. To conclude, an error analysis with the proposed error indicators is presented.

7.1 Sensors placement

In section 5, it was proposed how to obtain the location of the sensors in the framework of the EIM/DEIM technique. Note that, in case the sensors were already placed in the device, this step could be omitted. Figure 2 shows the position of a different number of sensors (3, 5, 10, 20 and 30). It can be observed that the sensors are located in the first quadrant of the domain (top-right zone). The solution shown in figure 1, in the area where the sensors are placed, contains most of the information since in the rest of the domain the
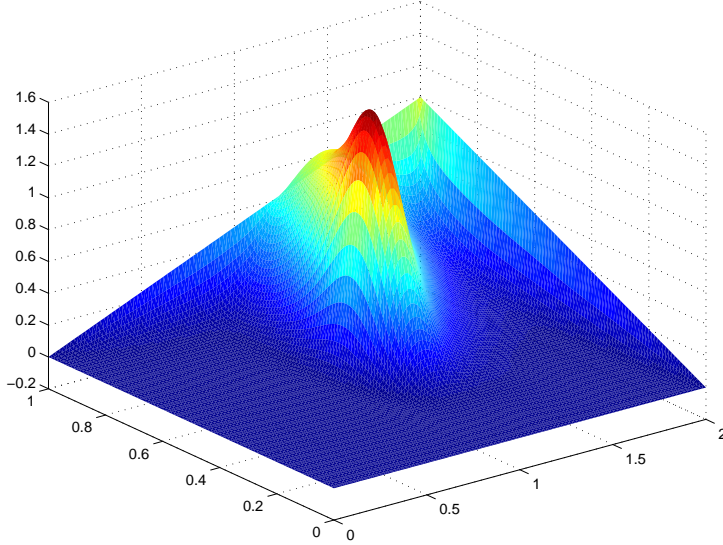
Figure 1: Reference solution obtained with the FEM for a diffusion coefficient $k_{\mathrm{ex}} = 0.037$.

solution is almost constant and close to zero. Note that the addition of a new sensor does not modify the location of the previous ones when considering the DEIM-based strategy.

Figure 3 shows the error between the Finite Element solution and the assimilated solution obtained with $\tau = 1$ for different number of sensors located at the positions dictated by the DEIM-based procedure. Figure 3 also considers the same number of sensors, but randomly distributed. In this case, 5 different random positions have been considered in each situation. The plot shows the mean value of the relative error of the assimilated solution. When considering 2 and 4 measures and the DEIM and random locations, the associated errors are denoted by $\mathcal{E}_{EX,FEM}^{2,\mathrm{DEIM}}$, $\mathcal{E}_{EX,FEM}^{4,\mathrm{DEIM}}$, $\mathcal{E}_{EX,FEM}^{2,\mathrm{Rand}}$ and $\mathcal{E}_{EX,FEM}^{4,\mathrm{Rand}}$. The figure shows that the error associated with the DEIM-based approach is smaller in all situations and moreover it is almost independent of the number of sensors when considering $\tau = 1$. However, when random positions are considered, the mean value of the error is higher than the value obtained when the sensor placement procedure is used, specially in the case of 2 sensors whose mean value remains almost constant. However, it is expected that the accuracy with the random approach improves for a high number of sensors.
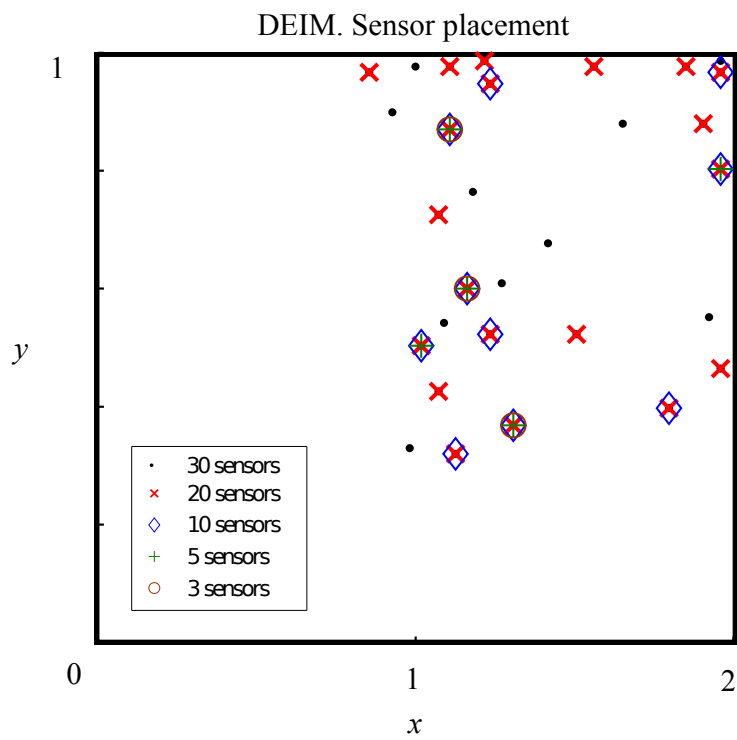
Figure 2: Placement of the sensors into the problem domain using the DEIM algorithm. The different number of sensors are represented with different marks.
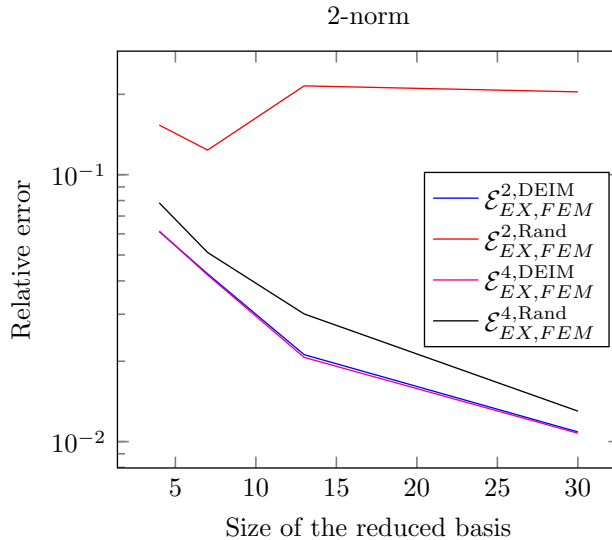
Figure 3: Relative error indicator associated with the difference between the Finite Element solution and the reduced solution for 2 and 4 sensors placed randomly (Rand) and dictated by the DEIM-based procedure (DEIM) for $\tau = 1$.
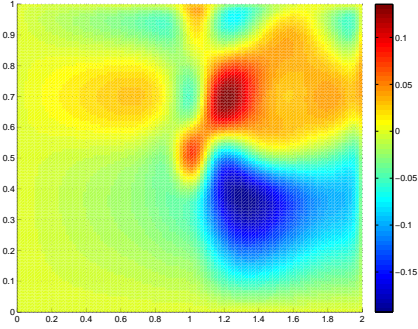
7.2 Reconstruction of the solution

Figure 4 shows the relative error between the FEM solution with the exact value of the parameter and the reconstructed solution with different size of the reduced basis and 9 sensors. It can be observed that with 5 terms the quality of the solution is acceptable, i.e. the maximum local error is smaller than 7 %. Note that the parameter inverse identification and the solution reconstruction are performed online and in real time. This is possible because the procedure only requires minimizing the functional defined in (15), which yields a small nonlinear system of equations with size $n_R + 1$, in which only one equation is nonlinear.

7.3 Convergence analysis

Figure 5 shows how the number of modes of the reduced solution increases when the prescribed error decreases. This value is obtained with expression (27), modifying the size of the reduced basis. This value will provide, offline, the number of modes that the reduced solution needs.
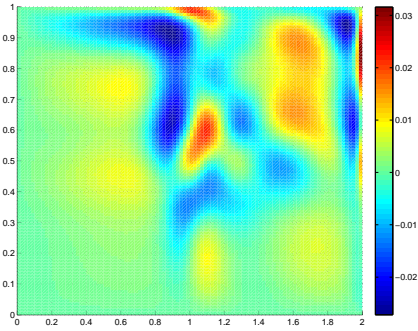
Figure 6 shows the error evolution (29) measured with the 2-norm and with the ∞-norm. It shows that the dependency of the solution error with the number of measures is small. This is one important advantage of the
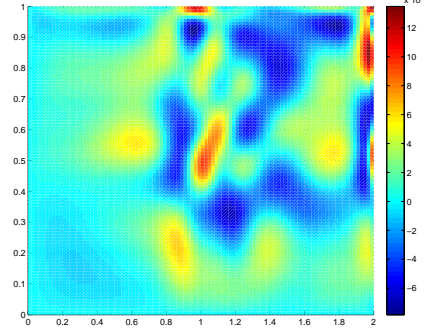
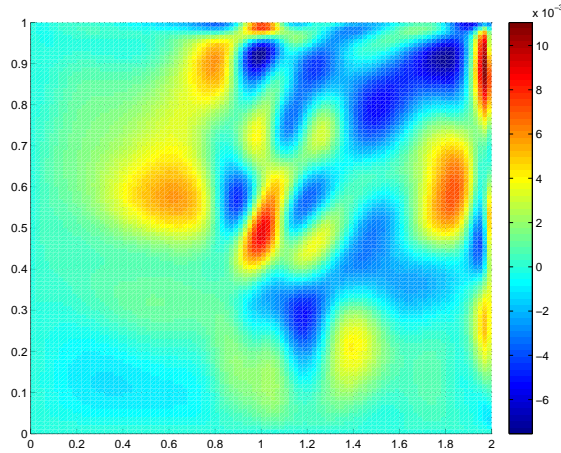(a) Reduced solution with 3 modes   (b) Reduced solution with 5 modes
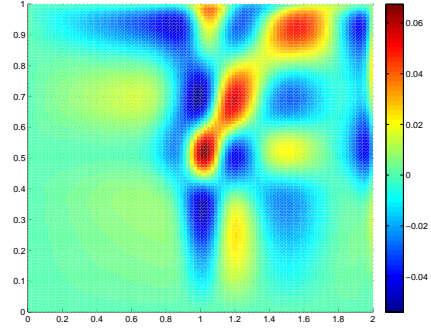
(c) Reduced solution with 10 modes   (d) Reduced solution with 20 modes

(e) Reduced solution with 30 modes

Figure 4: Relative error of the reconstructed solution with the reduced basis and the information of the sensors for different size of the reduced basis. The number of sensors is 9 and $\tau = 1$.
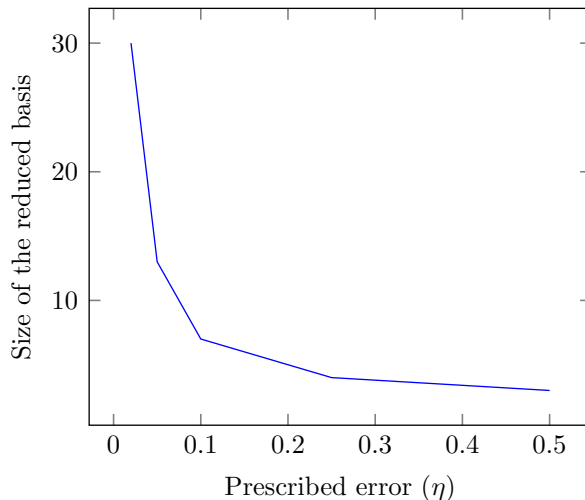
Figure 5: Evolution of the size of the reduced basis with prescribed error.

proposed method since it includes all the knowledge provided by the PGD solution in the minimization problem. Therefore, the results are accurate with only few measures.

Figure 7 show the error evolution (29), in 2-norm and in ∞-norm respectively, for different values of the parameter τ. It can be noticed that the size of the reduced basis has a considerable influence on both, the local and the global accuracy, however, the influence of τ is negligible. Note that the convergence of the PGD solution in the ∞-norm is not ensured [31], but the experimental results prove that it converges in this particular case.

Figure 8 shows the evolution of the error in the parameter identification with the size of the reduced basis. It clearly shows that, when the size increases, the accuracy in the inverse identification also increases for any value of $\tau \in [0.001, 9]$. It is important to notice that the accuracy of the error estimator is very high in the problem here addressed even considering few modes (3 or 5). However, as it can be clearly appreciated in the figure, for the firsts initial modes the error slightly increases. This fact is due to the value of k_A is obtained by minimizing (15). The minimization process tries to obtain simultaneously $\boldsymbol{\alpha}$ and k_A with the given data and this process does not strictly guarantee the monotonic convergence of only of one of them.

Figure 9 shows the effect of τ when considering a small number of sensors. In this case the size of the reduced basis is equal to 2. When the size of the basis is high enough, the effect of τ is negligible. Figure 9 also shows that when the number of sensors in high enough (around 10), the influence of τ is small. This is due to the fact that the information provided by the

21

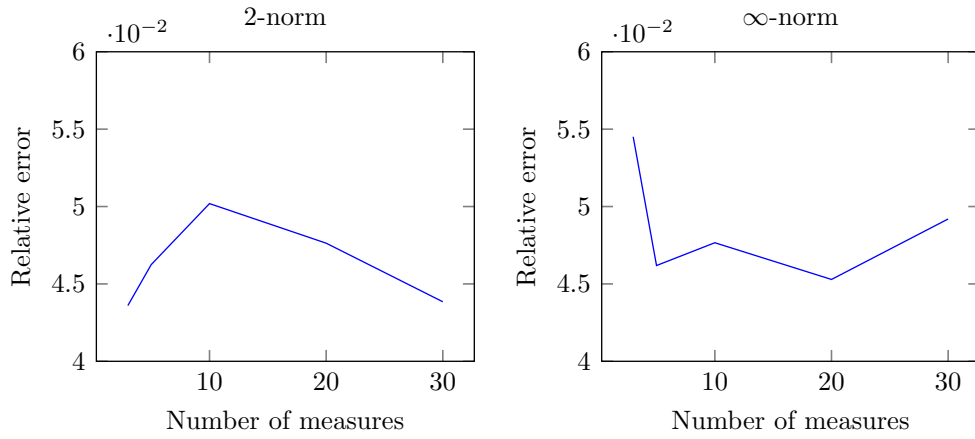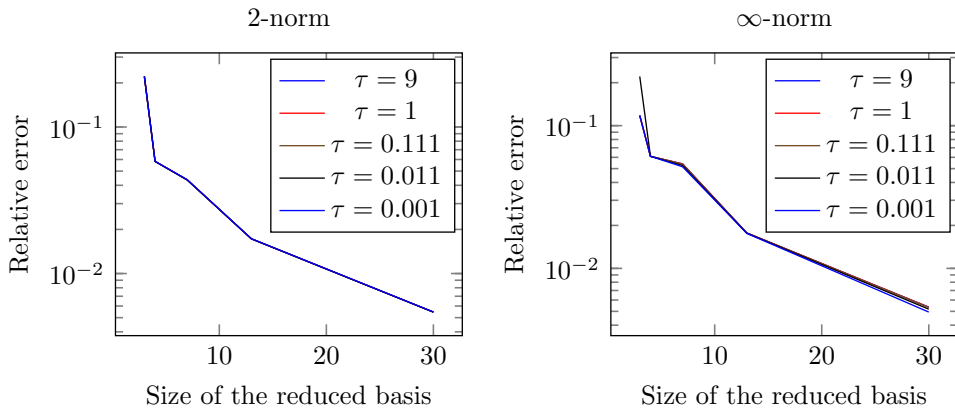Figure 6: Evolution of the error with the number of sensors with $n_R = 7$ and $\tau = 1$.

Figure 7: Evolution of the relative error in 2-norm and n ∞-norm with respect to the size of the reduced basis with respect to the full PGD solution, when considering 20 sensors. Note that the influence of τ is negligible.
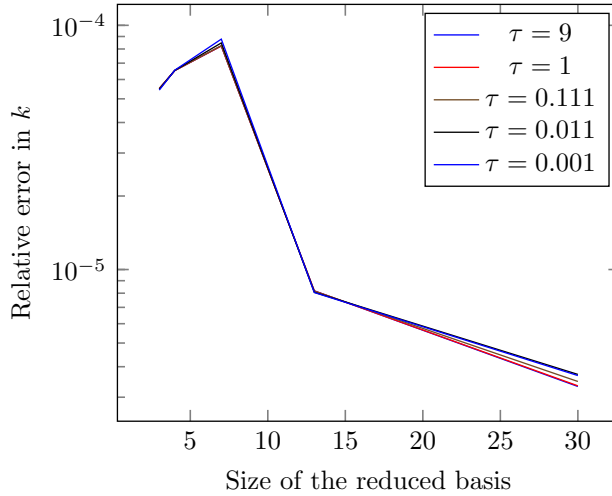
Figure 8: Relative error in the evaluation of the parameter whose exact value is $k_{\text{ex}} = 0.037$. 20 sensors are considered. Note that the influence of τ is negligible.

sensors is rich enough. However, when only few sensors are used (e.g. 2), the regularization plays an important role, and then the solution improves by increasing τ.

The final analysis is to check the quality of the proposed error upper bound measure. Figure 10 shows the evolution of the error upper bound (blue line) with respect to the size of the reduced basis. A comparison is also provided with the error indicator proposed in this work (red line). It can be observed that effectively the upper bound property holds. Note that the error upper bound is evaluated for all parameter values, thus sub-optimal upper error bounds are obtained. Additionally the exact error in 2-norm is also represented in brown. In particular the difference between the PGD solution with the exact value of the parameter (in brown) and the assimilated solution (29). Both curves almost overlap for a reasonable size of the reduced basis (in red), therefore the error indicator (30) is providing accurate error measures.
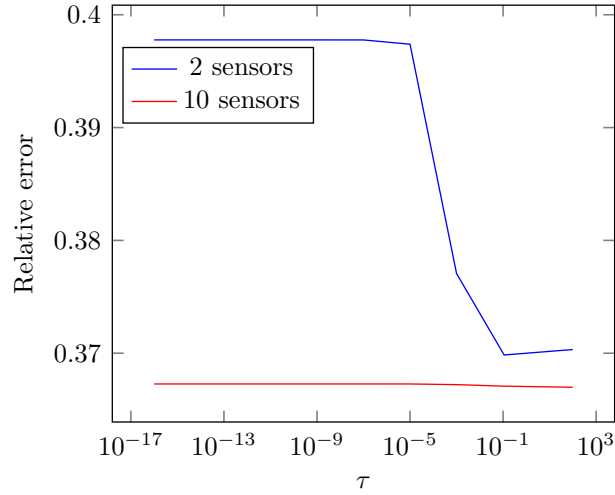
23

Figure 9: Evolution of the relative error in 2-norm of the reduced solution ($n_\mathrm{R} = 2$) for different values of τ, with respect to the full compact PGD solution.
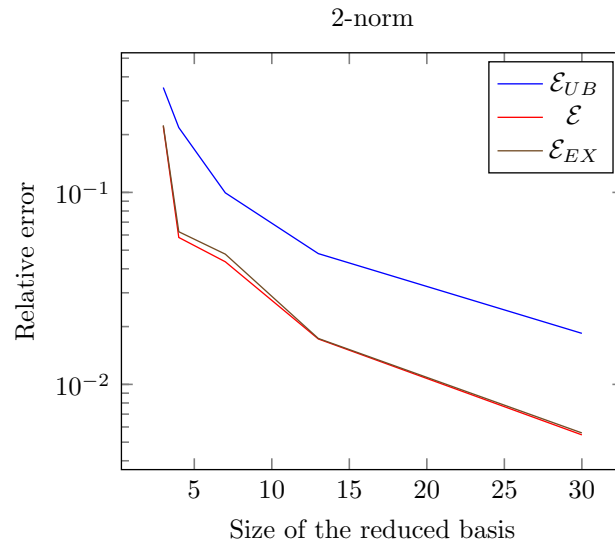

Figure 10: Analysis of the error upper bound. Comparison between the estimated error indicator $\mathcal{E}$ and the exact error $\mathcal{E}_{EX}$.

24

8 Conclusions

The techniques presented in this work are able to operate under the constraint of online decision making. These techniques allow to reconstruct in real-time the solution of a parametric partial differential equation from both, the data provided from the sensors and also from the mathematical model of the problem. Additionally, the current approach is also able to identify the model parameters, such as the diffusion coefficient, with a high level of accuracy. Therefore, these techniques could certainly account in real-time for the temporal evolution of material properties or any other parameter of the model. Consequently, this is a valuable tool to control loops of industrial processes aiming at preserving the quality of the products. Moreover, since the full PGD parametric solution is available, it is possible to determine, *a priori*, the size of the reduced basis for a certain level of accuracy. Furthermore, an error indicator in order to control the quality of the reconstructed solution is also introduced. The results prove that a small number of sensors is enough to fulfil efficiently both tasks: the inverse identification of model parameters and the solution reconstruction. Future research will aim at extending the present approach to deal with more realistic material models exhibiting nonlinear behavior and transient effects.

References

[1] Y. Cao, J. Zhu, I. M. Navon, and Z. Luo, "A reduced-order approach to four-dimensional variational data assimilation using proper orthogonal decomposition," *International Journal for Numerical Methods in Fluids*, vol. 53, pp. 1571–1583, 2007.

[2] D. N. Daescu and I. M. Navon, "Efficiency of a POD-based reduced second-order adjoint model in 4D-Var data assimilation," *International Journal for Numerical Methods in Fluids*, vol. 53, pp. 985–1004, 2007.

[3] I. Hoteit, D.-T. Pham, and J. Blum, "A simplified reduced order Kalman filtering and application to altimetric data assimilation in Tropical Pacific," *Journal of Marine Systems*, vol. 36, pp. 101–127, July 2002.

[4] S. S. Ravindran, "A reduced-order approach for optimal control of fluids using proper orthogonal decomposition," *International Journal for Numerical Methods in Fluids*, vol. 34, pp. 425–448, Nov. 2000.

[5] K. Willcox and J. Peraire, "Balanced Model Reduction via the Proper Orthogonal Decomposition," *AIAA journal*, vol. 40, pp. 2323–2330, May 2002.

[6] F. M. Selten, "Baroclinic Empirical Orthogonal Functions as Basis Functions in an Atmospheric Model," *Journal of the Atmospheric Sciences*, vol. 54, pp. 2099–2114, Aug. 1997.

[7] P. T. M. Vermeulen and A. W. Heemink, "Model-Reduced Variational Data Assimilation," *Monthly weather review*, vol. 134, pp. 2888–2899, 2006.

[8] R. Herzog and I. Riedel, "Sequential optimal sensor placement in thermoelastic models," *Technische Universitat Chemnitz*, vol. Preprint, 2013.

[9] A. A. Alonso, I. G. Kevrekidis, J. R. Banga, and C. E. Frouzakis, "Optimal sensor location and reduced order observer design for distributed process systems," *Computers & Chemical Engineering*, vol. 28, pp. 27–35, Jan. 2004.

[10] J. A. Atwell and B. B. King, "Proper Orthogonal Decomposition for Reduced Basis Feedback Controllers for Parabolic Equations," *Mathematical and Computer Modelling*, vol. 33, pp. 1–19, 2001.

[11] G. Chardon, A. Cohen, and L. Daudet, "Sampling and reconstruction of solutions to the Helmholtz equation.," *Sampling theory on signal and image processing*, vol. Submitted, 2014.

[12] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera, "Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 41, pp. 575–605, Aug. 2007.

[13] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, "An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations," *Comptes Rendus Mathematique*, vol. 339, pp. 667–672, Nov. 2004.

[14] Y. Maday, A. T. Patera, J. D. Penn, and M. Yano, "A Parametrized-Background Data-Weak Approach to Variational Data Assimilation: Formulation, Analysis, and Application to Acoustics," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 0, pp. 1–32, 2014.

[15] Y. Maday and O. Mula, "A generalized empirical interpolation method : application of reduced basis techniques to data assimilation," *Analysis and Numerics of Partial Differential Equations*, pp. 221–235, 2013.

[16] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings, "A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids," *Journal of Non-Newtonian Fluid Mechanics*, vol. 139, pp. 153–176, Dec. 2006.

[17] F. Chinesta, R. Keunings, and A. Leygue, *The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer.* Springer Publishing Company, Incorporated, Oct. 2013.

[18] F. Chinesta, A. Ammar, and E. Cueto, "Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models," *Archives of Computational Methods in Engineering*, vol. 17, no. 4, pp. 327–350, 2010.

[19] F. Chinesta, A. Leygue, F. Bordeu, J. V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar, and A. Huerta, "PGD-Based Computational Vademecum for Efficient Design, Optimization and Control," *Archives of Computational Methods in Engineering*, vol. 20, pp. 31–59, Jan. 2013.

[20] S. Chaturantabut and D. C. Sorensen, "Nonlinear Model Reduction via Discrete Empirical Interpolation," *SIAM Journal on Scientific Computing*, vol. 32, pp. 2737–2764, Jan. 2010.

[21] F. Chinesta, A. Ammar, A. Leygue, and R. Keunings, "An overview of the proper generalized decomposition with applications in computational rheology," *Journal of Non-Newtonian Fluid Mechanics*, vol. 166, no. 11, pp. 578–592, 2011.

[22] F. Chinesta, P. Ladeveze, and E. Cueto, "A Short Review on Model Order Reduction Based on Proper Generalized Decomposition," *Archives of Computational Methods in Engineering*, vol. 18, pp. 395–404, Oct. 2011.

[23] P. Ladevèze and L. Chamoin, "On the verification of model reduction methods based on the proper generalized decomposition," *Computer Methods in Applied Mechanics and Engineering*, vol. 200, pp. 2032–2047, June 2011.

[24] J. P. Moitinho de Almeida, "A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics," *International Journal for Numerical Methods in Engineering*, vol. 94, pp. 961–984, 2013.

[25] A. Ammar, F. Chinesta, P. Diez, and A. Huerta, "An error estimator for separated representations of highly multidimensional models," *Computer Methods In Applied Mechanics And Engineering*, vol. 199, no. 25-28, pp. 1872–1880, 2010.

[26] B. Bognet, F. Bordeu, F. Chinesta, A. Leygue, and A. Poitou, "Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity," *Computer Methods in Applied Mechanics and Engineering*, vol. 201-204, pp. 1–12, Jan. 2012.

[27] B. Bognet, A. Leygue, and F. Chinesta, "Separated representations of 3D elastic solutions in shell geometries," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 1, no. 1, pp. 1–4, 2014.

[28] C. Ghnatios, F. Chinesta, and C. Binetruy, "3D Modeling of squeeze flows occurring in composite laminates," *International Journal of Material Forming*, pp. 1–11, Oct. 2013.

[29] J. Donea and A. Huerta, *Finite element methods for flow problems.* J. Wiley and Sons, 2002.

[30] D. Gonzalez, E. Cueto, F. Chinesta, P. Diez, and A. Huerta, "SUPG-Based Stabilization of Proper Generalized Decompositions for High-Dimensional Advection-Diffusion Equations," *International Journal for Numerical Methods in Engineering*, vol. 94, no. 13, pp. 1216–1232, 2013.

[31] A. Falcó and A. Nouy, "Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces," *Numerische Mathematik*, vol. 121, pp. 503–530, Dec. 2011.