Model Reduction Methods

Francisco Chinesta[1], Antonio Huerta[2], Gianluigi Rozza[3] and Karen Willcox[4]

[1]*ESI GROUP, High Performance Computing Institute – ICI, École Centrale de Nantes, Nantes, France*
[2]*Laboratory of Computational Methods and Numerical Analysis, Polytechnic University of Catalunya, Barcelona, Spain*
[3]*Mathematics Area, mathLab, SISSA, International School for Advanced Studies, Trieste, Italy*
[4]*Center for Computational Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA*

1 INTRODUCTION

The human brain consumes 4 W for performing some tasks for which today's computers require the power of several nuclear plants. It is then clear that our computers and algorithms for addressing the models encountered in science and engineering are definitively suboptimal.

Many problems related to important society needs require fast and accurate solutions, in general data-driven, of complex models, involving an unimaginable amount of information, in many cases in real time and on deployed platforms. Up to now, the solution of complex models has been addressed by using high-performance and powerful computing platforms. In the future, real-time simulation, optimization, and control, in science and engineering, will require as much computational power (supercomputing) as possible, and consequently, advances in hardware and software for high-performance computing will be necessary. But at the same time, there is a need for a new generation of simulation techniques, beyond high-performance computing or modern solver approaches (many of them proposed 40 years ago), to improve efficiency or to allow getting results when other alternatives fail in the above challenging scenarios.

The importance of dynamic data-driven application systems (DDDASs) in the coming decades has been already noticed by the NSF Blue Ribbon Panel on Simulation Based Engineering Sciences report (Oden *et al.*, 2006), which in 2006 included DDDASs as one of the five core challenges in the field for the next decade (together with multiscale simulation, model validation and verification, handling large data, and visualization). This panel concluded that "Dynamic data-driven application systems will rewrite the book on the validation and verification of computer predictions" and that "research is needed to effectively use and integrate data-intensive computing systems, ubiquitous sensors and high-resolution detectors, imaging devices, and other data-gathering storage and distribution devices, and to develop methodologies and theoretical frameworks for their integration into simulation systems."

Encyclopedia of Computational Mechanics Second Edition,
Edited by Erwin Stein, René de Borst and Thomas J.R. Hughes.

DOI: 10.1002/9781119176817.ecm2110

1.1 Current-day computational issues

Today, many problems in science and engineering remain intractable, in spite of the impressive progresses attained in modeling, numerical analysis, discretization techniques, and computer science during the last decade. This is because their numerical complexity, or the restrictions imposed by different requirements (real-time on deployed platforms, for instance) make them unaffordable for today's technologies.

We illustrate this point by describing some of these problems:

- The first one concerns models that are defined in high-dimensional spaces, usually encountered in quantum chemistry describing the structure and mechanics of materials, the kinetic theory description of complex materials, social dynamics and economic systems, vehicular traffic flow phenomena, complex biological systems involving mutation and immune competition, and crowds and swarms encountered in congested and panic flows, and many other possibilities, such as the chemical modeling in very dilute systems where the concept of concentration cannot be used, which results in the so-called chemical master equation governing, for example, cell signaling and other phenomena in molecular biology. All these models involve D dimensions, $D \gg 3$, which limits the consideration of standard mesh-based discretizations.
- Online control can be carried out following different approaches. The most common one considers systems as a black box whose behavior is modeled by a transfer function relating inputs to outputs. This modeling approach, which may seem poor, has as its main advantage the possibility of proceeding rapidly due to its simplicity. This compromise between accuracy and speed was often used in the past, and this pragmatic approach has allowed us to control processes and to optimize them once the transfer function modeling the system is established. The establishment of such goal-oriented transfer function is the trickiest point. For this purpose, it is possible to proceed from a sometimes overly simplified physical model or directly from experiments (allowing us to extract a phenomenological, goal-oriented transfer function) or from a well-balanced mixture of both approaches. In all cases, the resulting modeling can only be applied within the framework that served to derive it. However, on one hand, the fine description of systems requires a sufficiently detailed description of them and, in that case, traditional goal-oriented simplified modeling becomes inapplicable. On the other hand, actual physical models result, in general, in complex mathematical objects, nonlinear and strongly coupled Partial Differential Equations (PDEs). Such mathematical objects represent physical reality up to a certain degree of accuracy. However, the available numerical tools capable of solving these complex models require the use of powerful computers that can require hours, days, and weeks to solve them. Known as numerical simulation, the resulting output solution is very rich but is unsuitable for control purposes that require fast responses, often in real time.
- Many problems in parametric modeling, inverse identification, and process or shape optimization usually require, when approached with standard techniques, the direct computation of a very large number of solutions of the concerned model for particular values of the problem parameters. When the number of parameters increases, such a procedure becomes intractable.
- Traditionally, simulation-based engineering sciences (SBESs) relied on the use of static data inputs to perform the simulations. These data could be parameters of the model(s) or boundary conditions. The word "static" is intended to mean here that these data could not be modified during the simulation. A new paradigm in the field of applied sciences and engineering has emerged in the last decade. DDDASs today constitute one of the most challenging applications of SBESs. By DDDAS we mean a set of techniques that allow the linkage of simulation tools with measurement devices for real-time control of simulations. DDDAS entails the ability to dynamically incorporate additional data into an executing application, and in reverse, the ability of an application to dynamically steer the measurement process.[1]
- Augmented reality is another area in which efficient (fast and accurate) simulation is urgently needed. The idea is to supply in real time appropriate information to the reality perceived by the user. Augmented reality could be an excellent tool in many branches of science and engineering. In this context, light computing platforms are appealing alternatives to heavy computing platforms which in general are expensive and whose use requires technical knowledge.
- In science and engineering, in its widest sense, there are many causes of variability. The introduction of such variability, randomness, and uncertainty is a priority for the next decade. Although it was a priority in the preceding decade, many open challenges remain.

While the previous list is by no means exhaustive, it includes a set of problems with no apparent relationship between them that can however be treated in a unified manner as will be shown in what follows. Their common ingredient is our lack of capabilities (or knowledge) to solve

them numerically in a direct, traditional but computationally tractable way.

1.2 Fast calculations from a historical perspective

The human being throughout history developed several facilities for giving fast responses to a variety of questions. Thus, abaci were used 2700 years BC in Mesopotamia. However, the initial arithmetic was rapidly complemented with more complex representations; some of them were the charts and the nomograms.

The former facilities allowed fast calculations and data manipulations. Nomograms can be easily constructed when the mathematical relationships that they express are purely algebraic, eventually nonlinear. In those cases, it was easy to represent some outputs as a function of some inputs. The calculation of these data representations was performed offline and then used online in many branches of engineering sciences for design and optimization.

However, the former procedures fail when addressing more complex scenarios. Thus, sometimes engineers manipulate not properly understood physics, and in that case the construction of nomograms based on too coarse a modeling could be dangerous. In those cases, one could proceed by conducting several experiments from which a sort of experiment-based nomogram could be defined. In other cases, the mathematical object to be manipulated consists of a system of complex coupled nonlinear PDEs, whose solution for each possible combination of the values of the parameters that it involves is simply unimaginable for modern computational availabilities. In these cases, experiments or expensive computational solutions are performed for some possible states of the system, from which a simplified model linking the inputs to the outputs of interest is elaborated. These simplified models have different names: surrogate models, metamodels, response surface methodologies, and so on. Other associated tricky questions include: what is the best sampling strategy (Latin hypercube, etc.), and what are the appropriate interpolation techniques for estimating the response at an unmeasured position from observed values at surrounding locations. Many possibilities exist, kriging being one of the most widely used for interpolating data. All these techniques allow defining a "numerical or graphical handbook." One of the earliest and most widely known within engineering practice is that of Bernoulli (1836).

Recently, model order reduction opened new possibilities. In this chapter, we describe three different model reduction techniques. First, proper orthogonal decomposition (POD) is a general technique for extracting the most significant characteristics of a system's behavior and representing them in a set of "POD basis vectors." These basis vectors then provide an efficient (typically low-dimensional) representation of the key system behavior, which proves useful in a variety of ways. The most common use is to project the system governing equations onto the reduced-order subspace defined by the POD basis vectors. This yields an explicit POD reduced model that can be solved in place of the original system. The POD basis can also provide a low-dimensional description in which to perform parametric interpolation, infill missing or "gappy" data, and perform model adaptation. There is extensive literature on this, and POD has seen broad application across fields. Some review of POD and its applications can be found in Dowell and Hall (2001), Volkwein (2011), and Benner *et al.* (2015). POD's foundations and main methodological elements are described in Section 2.

Another family of model reduction techniques lies in the use of reduced basis constructed by combining a greedy algorithm and *a posteriori* error indicators. As for the POD, the reduced basis method requires some amount offline work, but then the reduced basis model can be used online for solving different models with control of the solution accuracy, because the availability of error bounds. When the error is unacceptably high, the reduced basis can be enriched by invoking a greedy adaption strategy. This technique will be described in detail in Section 3. Useful review works on the subject are Rozza *et al.* (2008), Patera and Rozza (2007), Quarteroni *et al.* (2011), Manzoni *et al.* (2012), Rozza (2014), and Hesthaven *et al.* (2015).

Finally, we address in Section 4 techniques based on the use of separated representations, at the heart of the so-called proper generalized decomposition (PGD) methods. Such separated representations are considered when solving at-hand PDEs by employing procedures based on the separation of variables; then they were considered in quantum chemistry for approximating multidimensional quantum wave function, for example, Hartree–Fock and post-Hartree–Fock methods (Cancès *et al.*, 2003). In the 1980s, Pierre Ladeveze proposed the use of space-time separated representations of transient solutions involved in strongly nonlinear models, defining a nonincremental integration procedure (Ladeveze, 1985). Later, separated representations were employed for solving multidimensional models suffering the so-called curse of dimensionality (Ammar *et al.*, 2006, 2007), as well as in the context of stochastic modeling (Nouy, 2007). Then, they were extended for separating space coordinates, making possible the solution of models defined in degenerated domains, for example, plate and shells (Bognet *et al.*, 2012, 2014) as well as for addressing parametric models where model parameters were considered as model extra-coordinates, making possible the offline calculation of the parametric solution that can be viewed as a metamodel or a computational vademecum, to be used online for real-time simulation,

optimization, inverse analysis, and simulation-based control (see Chinesta *et al.*, 2013b, for a recent review).

Some recent reviews concerning PGD can be found in Chinesta *et al.* (2010b, 2011b), or Chinesta *et al.* (2011a), as well as in the recently published primer by Chinesta *et al.* (2013a).

In what follows we start by describing the POD, then we continue with the RB techniques, to finish the numerical journey by describing the main ideas and accomplishments of the PGD. Because the different techniques have been developed independently, sometimes the notations are not the same in the different sections of the present chapter. As the interested reader is generally addressed to the referred works on the different topics (POD, RB, and PGD), authors prefer to keep the usual notation that they employed in their previous works to facilitate the passage from this chapter to the many works already published on the different topics. Obviously, we took care of defining the notation employed in the different parts of this chapter when they were different from one section to another to avoid confusion. Certainly, the maturity of model order reduction techniques will be accompanied by a notational unification, so it is expected that future editions of this chapter will contain a unified notation.

In the same way, because authors in their respective former works illustrated the concepts and applicative developments from numerous images, schemas, and so on, all them available in the works referred to in the present chapter, it was decided to avoid figures and images duplicating the already published ones. For more details and illustrations on the techniques and their applications, the interested reader can refer to the numerous references and those within them.

2 PROPER ORTHOGONAL DECOMPOSITION

2.1 Introduction

POD is a method to compute a basis that provides a low-dimensional representation of a high-dimensional system state. POD was introduced for the analysis of turbulent flows by Lumley (1967), and is closely related to methods used in other fields such as Karhunen–Loéve expansions in stochastic process modeling (Loéve, 1955; Kosambi, 1943), principal component analysis in statistical analysis (Hotelling, 1933), and empirical orthogonal eigenfunctions in atmospheric modeling (North *et al.*, 1982). POD basis vectors are computed empirically using sampled data collected over a range of relevant system dynamics, typically using the method of snapshots, introduced by Sirovich (1987).

Because of its broad applicability to linear and nonlinear systems, POD has become widely used in many different application domains. The most common use is to project the system governing equations onto the reduced-order subspace defined by the POD basis vectors. This yields an explicit POD reduced model that can be solved in place of the original system. The POD basis can also provide a low-dimensional description in which to perform parametric interpolation, infill missing or "gappy" data, and perform model adaptation.

In this section, we first describe computation of the POD basis via the method of snapshots in Section 2.2. Section 2.3 then discusses construction of a POD reduced model. Section 2.4 describes dynamic data-driven adaptation of POD reduced models, and Section 2.5 describes the gappy POD.

2.2 Computing the POD basis: method of snapshots

Consider the general full system model

$$\boldsymbol{E}(\boldsymbol{\mu})\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} + \boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{u} = \boldsymbol{f}(\boldsymbol{\mu}, \boldsymbol{u}) \tag{1}$$

where $\boldsymbol{u} \in \mathbb{R}^n$ is the state vector of dimension n, and $\boldsymbol{\mu} \in \boldsymbol{D} \subset \mathbb{R}^P$ is a vector of P parameters. The full model operators are $\boldsymbol{A}(\boldsymbol{\mu}) \in \mathbb{R}^{n\times n}$, $\boldsymbol{E}(\boldsymbol{\mu}) \in \mathbb{R}^{n\times n}$, and $\boldsymbol{f}(\boldsymbol{\mu}, \boldsymbol{u}) \in \mathbb{R}^n$. We write the full model as a system of nonlinear ordinary differential equations (ODEs) for ease of exposition and to emphasize the general applicability of the POD methodology, but note that models of interest often (but not always) arise from the discretization of PDEs.

For many applications, particular output quantities are of interest, and it is the goal of model reduction to achieve an accurate mapping between the input parameters and these quantities of interest. We define $\boldsymbol{y} \in \mathbb{R}^q$ to be the q output quantities of interest, defined by a general mapping from parameters and states to outputs:

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{\mu}, \boldsymbol{u}) \tag{2}$$

Next we consider a set of n_s snapshots, $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_{n_s}$, which are state solutions computed at different instants in time and/or different parameter values. Here, $\boldsymbol{u}_j = \boldsymbol{u}(t_j; \boldsymbol{\mu}_j) \in \mathbb{R}^n$ denotes the jth snapshot, where t_j and $\boldsymbol{\mu}_j$ are, respectively, the time and parameter values for the jth snapshot. Define the snapshot matrix $\boldsymbol{U} \in \mathbb{R}^{n\times n_s}$, which contains the snapshot $\boldsymbol{u}_j$ as its jth column. The (thin) singular value decomposition (SVD) of $\boldsymbol{U}$ is written as

$$\boldsymbol{U} = \boldsymbol{X}\boldsymbol{\Sigma}\boldsymbol{Y}^{\mathrm{T}} \tag{3}$$

where the columns of the matrices $\boldsymbol{X} \in \mathbb{R}^{n\times n_s}$ and $\boldsymbol{Y} \in \mathbb{R}^{n_s\times n_s}$ are the left and right singular vectors of $\boldsymbol{U}$, respectively. $\boldsymbol{\Sigma} \in \mathbb{R}^{n_s\times n_s} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{n_s})$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{n_s} \geq 0$, are the singular values of $\boldsymbol{U}$, referred to as the POD singular values. The POD basis, $\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r]$, is then defined as the r left singular vectors of $\boldsymbol{U}$ that correspond to the r largest POD singular values. This yields an orthonormal basis.

POD provides an efficient low-dimensional representation of the snapshot data: among all orthonormal bases of size r, the POD basis minimizes the least squares error of snapshot reconstruction

$$\min_{\boldsymbol{V}\in\mathbb{R}^{n\times r}} ||\boldsymbol{U} - \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{U}||_F^2 = \min_{\boldsymbol{V}\in\mathbb{R}^{n\times r}} \sum_{i=1}^{n_s} ||\boldsymbol{u}_i - \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{u}_i||_2^2 = \sum_{i=r+1}^{n_s} \sigma_i^2 \tag{4}$$

The square of the error in snapshot representation is given by the sum of the squares of the singular values corresponding to those left singular vectors not included in the POD basis. Thus, the singular values provide quantitative guidance for choosing the size of the POD basis. A typical approach is to choose r so that

$$\frac{\sum_{i=1}^{r} \sigma_i^2}{\sum_{i=1}^{n_s} \sigma_i^2} > \kappa \tag{5}$$

where κ is a user-specified tolerance, often taken to be 99.9% or greater. The left-hand side of (5) is often referred to as the relative "energy" captured by the POD modes.

Since the POD basis is constructed from sampled solutions, the POD method makes no assumptions about the form of the full model – POD applies to both linear and nonlinear systems, as well as to parametrically varying systems. One can also include sensitivity information in the snapshot set, which may be advantageous in some settings (Hay *et al.*, 2009; Hinze and Volkwein, 2008).

For linear time-invariant (LTI) systems, one may also derive the POD in the frequency domain (Kim, 1998). In the case of a single-input single-output LTI system, the full model can be written as

$$\boldsymbol{E}\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} + \boldsymbol{A}\boldsymbol{u} = \boldsymbol{B}\mu, \quad y = \boldsymbol{C}\boldsymbol{u} \tag{6}$$

where the parameter is now a scalar $\mu(t)$, is time-dependent, and enters only through the forcing term $\boldsymbol{B}\mu$, where $\boldsymbol{B} \in \mathbb{R}^{n\times 1}$ is the input forcing vector (i.e., in the LTI case the matrices $\boldsymbol{A}$ and $\boldsymbol{E}$ do not depend on μ). The scalar output y is a linear combination of states, as defined by the output vector $\boldsymbol{C} \in \mathbb{R}^{1\times n}$.

The frequency-domain formulation of (6) is obtained by considering sinusoidal forcing $\mu = \overline{\mu}\mathrm{e}^{i\omega t}$ with frequency ω, leading to the complex frequency-domain equations

$$(i\omega\boldsymbol{E} + \boldsymbol{A})\overline{\boldsymbol{u}} = \boldsymbol{B}\overline{\mu}, \quad \overline{y} = \boldsymbol{C}\overline{\boldsymbol{u}} \tag{7}$$

where $\boldsymbol{u} = \overline{\boldsymbol{u}}\mathrm{e}^{i\omega t}$ and $y = \overline{y}\mathrm{e}^{i\omega t}$. Thus, another way to generate POD snapshots in the LTI setting is to sample frequencies $\omega_1, \omega_2, \ldots, \omega_{n_s}$ and compute the corresponding (complex) snapshots:

$$\overline{\boldsymbol{u}}_i = (i\omega_i\boldsymbol{E} - \boldsymbol{A})^{-1}\boldsymbol{B}, \quad i = 1, 2, \ldots, n_s \tag{8}$$

As discussed in Willcox *et al.* (2003), this view of POD shares similarity with the rational interpolation model reduction methods that employ Krylov subspaces (see Bai, 2002; Freund, 2003; Antoulas, 2005; Schilders *et al.* (2008), Antoulas *et al.* (2010), and Baur *et al.* (2011)).

In the linear setting, duality between time and frequency domain formulations reveals the connections between POD and balanced truncation (Lall *et al.*, 2002; Willcox and Peraire, 2002). In particular, if the POD snapshots are generated by simulating the system impulse response, one can interpret the POD as an approximation by quadrature of the reachability Gramian, and the POD basis vectors can be shown to approximate the most reachable modes in the system. Based on this observation, Willcox and Peraire (2002) proposed an approximate balanced truncation approach using the POD method of snapshots. The dual (or adjoint) system of (7) is

$$(i\omega\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{A}^{\mathrm{T}})\overline{\boldsymbol{z}} = \boldsymbol{C}^{\mathrm{T}}\overline{\mu}, \quad \overline{y} = \boldsymbol{B}^{\mathrm{T}}\overline{\boldsymbol{z}} \tag{9}$$

where $\boldsymbol{z}$ is the dual state (or adjoint vector). The ith dual snapshot $\overline{\boldsymbol{z}}_i$ computed at sample frequency ω_i is then given by

$$\overline{\boldsymbol{z}}_i = (i\omega_i\boldsymbol{E}^{\mathrm{T}} - \boldsymbol{A}^{\mathrm{T}})^{-1}\boldsymbol{C}^{\mathrm{T}} \tag{10}$$

Just as the standard POD basis vectors can be interpreted as approximating the most reachable modes in the system, computing a set of dual POD basis vectors using the dual snapshots (10) extracts the most observable modes in the system. Taking this idea a step further, an appropriate combination of primal and dual snapshots in a "balanced POD" approximation can be interpreted as an approximate balanced truncation (Lall *et al.*, 2002; Willcox and Peraire, 2002; Rowley, 2005).

2.3 POD reduced models

In classical model reduction approaches, POD reduced models are derived and used with an offline–online strategy. The offline phase involves (i) expensive simulations of the full model to generate the snapshots needed to compute the

POD basis, (ii) projection of the full model onto the reduced subspace defined by the POD basis, and (iii) interpolation or some other strategy to efficiently represent parametric dependence and/or nonlinear terms. Step (i) was described in Section 2.2; here we discuss steps (ii) and (iii). The online phase is where the reduced model is employed to achieve rapid computations.

Derivation of the POD reduced model begins by approximately representing the full state in the POD basis:

$$\boldsymbol{u} \approx \boldsymbol{V}\boldsymbol{u}_r \tag{11}$$

where $\boldsymbol{u}_r \in \mathbb{R}^r$ is the reduced state vector containing the POD modal coefficients. Substituting the approximation (11) into the full model equations leads to a residual, since in general the true state will not lie exactly in the span of the POD basis. A Galerkin projection imposes orthogonality between the residual and the POD basis.

We first consider the LTI system (6). The Galerkin projection of the residual leads to the reduced model

$$\boldsymbol{E}_r \frac{\mathrm{d}\boldsymbol{u}_r}{\mathrm{d}t} + \boldsymbol{A}_r\boldsymbol{u}_r = \boldsymbol{B}_r\mu, \quad y_r = \boldsymbol{C}_r\boldsymbol{u}_r \tag{12}$$

where the reduced model operators are $\boldsymbol{A}_r = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{V} \in \mathbb{R}^{r\times r}$, $\boldsymbol{E}_r = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{E}\boldsymbol{V} \in \mathbb{R}^{r\times r}$, $\boldsymbol{B}_r = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{B} \in \mathbb{R}^{r\times 1}$, and $\boldsymbol{C}_r = \boldsymbol{C}\boldsymbol{V} \in \mathbb{R}^{1\times r}$. y_r is the output of the reduced model, which approximates the full model output y. While a Galerkin projection is the most commonly used, a Petrov–Galerkin projection is also possible and is desirable in some settings (e.g., to preserve stability). A Petrov–Galerkin projection requires definition of a left POD basis, $\boldsymbol{W} \in \mathbb{R}^{n\times r}$, using, for example, the balanced POD approach. In this case, the POD reduced model operators become $\boldsymbol{A}_r = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{V}$, $\boldsymbol{E}_r = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{E}\boldsymbol{V}$, and $\boldsymbol{B}_r = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{B}$ (with the same dimensions as above and with $\boldsymbol{C}_r$ unchanged).

In the case of the general nonlinear system (1), one proceeds in the same way to derive the POD reduced model, yielding the reduced system

$$\boldsymbol{E}_r(\boldsymbol{\mu})\frac{\mathrm{d}\boldsymbol{u}_r}{\mathrm{d}t} + \boldsymbol{A}_r(\boldsymbol{\mu})\boldsymbol{u}_r = \boldsymbol{f}_r(\boldsymbol{\mu}, \boldsymbol{u}_r) \tag{13}$$

where the nonlinear term is

$$\boldsymbol{f}_r(\boldsymbol{\mu}, \boldsymbol{u}_r) = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mu}, \boldsymbol{V}\boldsymbol{u}_r) \tag{14}$$

While $\boldsymbol{f}_r$ is of reduced dimension, (14) shows that its evaluation requires computations that involve the large dimension n, making the reduced model expensive to solve. To address this issue, one must introduce an additional approximation that removes the direct dependence of $\boldsymbol{f}_r$ on $\boldsymbol{V}\boldsymbol{u}_r$. The most successful methods to achieve this combine selective spatial sampling of $\boldsymbol{f}$ with an interpolation strategy. Among this class of methods, the missing point estimation (Astrid *et al.*, 2008) and Gauss Newton with approximated tensors (GNATs) (Carlberg *et al.*, 2013) methods both build upon the gappy POD interpolation method (Everson and Sirovich, 1995), while the empirical interpolation method (EIM) of Barrault *et al.* (2004) and its discrete variant, the discrete empirical interpolation method (DEIM) of Chaturantabut and Sorensen (2010), conduct interpolation on a low-dimensional basis for the nonlinear term. We refer to the cited papers for a detailed discussion of these methods.

Similarly, one must address efficient handling of reduced model parametric dependence. Terms in (13) such as $\boldsymbol{A}_r(\boldsymbol{\mu}) = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{A}(\boldsymbol{\mu})\boldsymbol{V}$ could prevent the offline–online decomposition, since for every new value of $\boldsymbol{\mu}$ one would need to evaluate the full system matrix $\boldsymbol{A}(\boldsymbol{\mu})$ and then project it to obtain $\boldsymbol{A}_r(\boldsymbol{\mu})$, making the evaluation of the reduced model in the online phase expensive. Strategies to address this typically involve using one of the interpolation methods mentioned previously, possibly combined with a method that interpolates POD bases computed at different parameter points (Amsallem and Farhat, 2008). See Benner *et al.* (2015) for an in-depth discussion of the various ways of handling parametric dependence in reduced models.

Accuracy of POD reduced models is an important question for which rigorous analysis can be applied in only a limited setting. It is important to note that the error metric (4) applies to the error in the reconstruction of the snapshots, not to the error in solution of the POD reduced model (i.e., not to the error $y_r - y$). However, derivation of error estimates for POD reduced models is possible in some settings, depending on the properties of the underlying full system equations, just as for the reduced basis method error estimates presented in Section 3.8. See Rathinam and Petzold (2003), and Hinze and Volkwein (2005) for analysis of the error of POD reduced models.

It is also important to note that the choice of the snapshot set clearly impacts the POD basis and thus the POD reduced model. Optimal snapshot selection for the case of a time-dependent system but no parametric dependence is considered in Kunisch and Volkwein (2010), where the time locations of snapshots are chosen to minimize the error between the POD solution and the trajectory of the original dynamical system. In many cases, POD is used to create a reduced model that targets a particular range of system behavior; in those cases, the snapshot set is chosen based on thorough sampling of the full system model over the desired range of validity of the reduced model. However, if the system depends on more than two or three parameters, it may not be feasible to generate snapshots by brute-force sampling. In these cases, one can employ sparse grid sampling or greedy sampling, described in Section 3.6.

For example, Bui-Thanh *et al.* (2008) use greedy sampling to derive a POD reduced model for a thermal problem with 21 parameters.

2.4 Dynamic data-driven adaptation via POD

As mentioned previously, classical POD model reduction follows an offline–online strategy, where the reduced model is built once in the offline phase and then used in the online phase. However, in many applications of interest, data may become available in the online phase. For example, real-time decision support is one attractive application area for model reduction. The DDDAS paradigm, discussed in the introduction, is one particular setting where online data are available and could be used to adapt the reduced model. In real-time decision making in the context of DDDAS, the involved models and computational methods have to meet two particular requirements. First, a decision has to be made in real time (or near real time) and thus the model runtime must be short. Second, as the underlying dynamic system changes, the model must adapt to the changed system by learning from data generated by sensors. The adapted model in turn allows steering of the data-gathering process, which closes the DDDAS loop.

Model reduction offers a solution to the real-time constraint; however, classical model reduction with the offline–online decomposition is contrary to the DDDAS paradigm, which explicitly requires the dynamic adaptation of the reduced model with the new data received from the sensors during the online phase. In Peherstorfer and Willcox (2015a), dynamic reduced models are proposed, which break with the classical but rigid splitting into offline and online phase and allow the online adaptation of the reduced model. Dynamic reduced models take into account changes in the underlying system, by directly learning from the data provided by the sensors. To be useful, this adaptation must be achieved rapidly, without recourse to the computationally expensive full model.

Consider a system in the context of DDDAS that depends on an observable parameter $\boldsymbol{\mu} \in \mathcal{D}$ and on a latent parameter $\boldsymbol{\eta} \in \mathcal{E}$. The observable parameter $\boldsymbol{\mu}$ describes known properties of the system, such as material properties and load. In contrast, the latent parameter $\boldsymbol{\eta}$ describes unknown properties, such as damage, fatigue, and erosion. The value of the latent parameter is unknown, except for a nominal latent parameter $\boldsymbol{\eta}_0 \in \mathcal{E}$ (e.g., "no-damage" state of the system).

Consider a full model that describes a system with observable and latent parameters

$$\boldsymbol{A}_{\boldsymbol{\eta}}(\boldsymbol{\mu})\boldsymbol{u}_{\boldsymbol{\eta}}(\boldsymbol{\mu}) = \boldsymbol{f}(\boldsymbol{\mu}) \tag{15}$$

For ease of exposition, we consider here a full model that is linear in the state $\boldsymbol{u}_{\boldsymbol{\eta}}(\boldsymbol{\mu})$ but can depend nonlinearly on the observable $\boldsymbol{\mu}$ and latent parameter $\boldsymbol{\eta}$. We also consider a steady system (i.e., no transient system dynamics, although note that the latent parameter $\boldsymbol{\eta}$ is permitted to change over time). In (15), the dependence of the full model operator $\boldsymbol{A}$ and the state $\boldsymbol{u}$ on the latent parameter is denoted via a subscript, since that parametric dependence is treated differently to the setting described above for $\boldsymbol{\mu}$. In particular, since the value of $\boldsymbol{\eta}$ is unknown online, the latent parameter $\boldsymbol{\eta}$ cannot be incorporated into a reduced model as just another observable parameter.

We can use the standard POD framework of Sections 2.2 and 2.3 to build a POD basis and reduced model for the full model (15) defined at the nominal conditions. We denote the nominal (known) latent parameter by $\boldsymbol{\eta}_0$. To construct a reduced model for $\boldsymbol{\eta} = \boldsymbol{\eta}_0$, the full model is sampled with respect to the observable parameter at values $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_m \in \mathcal{D}$ and with constant $\boldsymbol{\eta} = \boldsymbol{\eta}_0$. This generates the m snapshots $\boldsymbol{U} = [\boldsymbol{u}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}_1), \ldots, \boldsymbol{u}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}_m)] \in \mathbb{R}^{n \times m}$. An r-dimensional POD basis $\boldsymbol{V}_{\boldsymbol{\eta}_0} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r] \in \mathbb{R}^{n \times r}$ is derived from $\boldsymbol{U}$, and the reduced model

$$\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu})\tilde{\boldsymbol{u}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) = \tilde{\boldsymbol{f}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) \tag{16}$$

of the full model (15) is constructed via Galerkin projection. The reduced operator is $\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) \in \mathbb{R}^{r \times r}$, the reduced state is $\tilde{\boldsymbol{u}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) \in \mathbb{R}^r$, and the reduced right-hand side is $\tilde{\boldsymbol{f}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) \in \mathbb{R}^r$.

The reduced model (16) is valid only for the latent parameter $\boldsymbol{\eta} = \boldsymbol{\eta}_0$, and therefore if $\boldsymbol{\eta}$ changes to, say, $\boldsymbol{\eta}_1 \in \mathcal{E}$, the reduced model (16) becomes obsolete. Recall that we are in the setting of DDDAS and receive sensor data of the underlying system that informs us (indirectly) about changes in $\boldsymbol{\eta}$. This means that if the latent parameter changes to $\boldsymbol{\eta} = \boldsymbol{\eta}_1$, we receive the m' pieces of sensor data $\boldsymbol{S} = [\hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+1}), \ldots, \hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+m'})] \in \mathbb{R}^{n \times m'}$, where $\boldsymbol{\mu}_{m+1}, \ldots, \boldsymbol{\mu}_{m+m'} \in \mathcal{D}$. The sensor samples $\hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+i})$ are approximations of the states $\boldsymbol{u}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+i})$, which might be corrupted with noise, for $i = 1, \ldots, m'$.

One possibility to adapt the reduced model would be to infer the latent parameter from the sensor data $\boldsymbol{S}$ using techniques from inverse problems, to assemble the corresponding full model, and, finally, to construct a new reduced model from scratch; however, this is often computationally too expensive in the context of real-time decision making and DDDAS. Instead, the dynamic reduced modeling approach introduced in Peherstorfer and Willcox (2015a) directly adapts the reduced model with the sensor data, without inferring the latent parameter and without recourse to the full model.

The adaptation of dynamic reduced models proceeds in two steps. First, the POD basis $\boldsymbol{V}_{\boldsymbol{\eta}_0}$ is adapted to $\boldsymbol{V}_{\boldsymbol{\eta}_1}$. For

that, the snapshots in $\boldsymbol{U}$ are replaced by sensor samples in $\boldsymbol{S}$. Since each replacement is a rank-1 update to $\boldsymbol{U}$, the new basis can be computed with runtime costs scaling only linearly with the number of degrees of freedom n of the full model (Brand, 2006). In the second step, an additive update $\delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}) \in \mathbb{R}^{r\times r}$ is constructed to obtain the adapted reduced operator $\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) + \delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}$. The update $\delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu})$ is the solution of the minimization problem

$$\underset{\delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu})\in\mathbb{R}^{r\times r}}{\arg\min} \sum_{i=m+1}^{m+m'} \|(\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) + \delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}))\boldsymbol{V}_{\boldsymbol{\eta}_1}^{\mathrm{T}}\hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_i) - \boldsymbol{V}_{\boldsymbol{\eta}_1}^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{\mu}_i)\|_2^2 \tag{17}$$

where $\hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+1}), \ldots, \hat{\boldsymbol{u}}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu}_{m+m'}) \in \mathbb{R}^r$ are the sensor samples. If sufficiently many sensor samples are available, and if the sensor samples are noise-free, then the adapted reduced operator $\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_0}(\boldsymbol{\mu}) + \delta\tilde{\boldsymbol{A}}_{\boldsymbol{\eta}_1}$ equals the true reduced operator that would be obtained by rebuilding the reduced model from scratch. The computational procedure derived in Peherstorfer and Willcox (2015a) splits (17) into r independent minimization problems that can then be solved in parallel.

The dynamic reduced model is adapted directly with the sensor samples. The adaptation avoids the computationally expensive inference of the value of the latent parameter, and it avoids the assembly of the large-scale operator $\boldsymbol{A}_{\boldsymbol{\eta}_1}(\boldsymbol{\mu})$ of the full model for the changed latent parameter $\boldsymbol{\eta}_1$. The adaptation of a dynamic reduced model can therefore be orders of magnitude faster than rebuilding a reduced model from scratch.

Similar dynamic data-driven strategies can be used to adapt nonlinear reduced models. The localized discrete empirical interpolation method (LDEIM) of Peherstorfer *et al.* (2014) uses machine learning techniques to construct multiple local DEIM interpolants in the offline phase, each tailored to a specific system behavior. In the online phase, the method switches between these local interpolants as the computation proceeds. This keeps the dimension of the local DEIM spaces, and thus the computational costs, low, while still approximating well a wide range of different system behaviors. In Peherstorfer and Willcox (2015b), an online adaptive DEIM constructs a reduced model with POD and the DEIM in the offline phase, and then adapts the DEIM interpolation points and the basis of the DEIM space with low-rank updates during the online phase. In each adaptivity step, an update is derived from sparse samples of the full model. The update minimizes the residual of the DEIM approximation in the Frobenius norm in each step.

2.5 Gappy POD

The gappy POD is a variant of POD that considers missing or "gappy" data. This procedure was developed by Everson and Sirovich (1995) in the context of facial image reconstruction and was first applied to reconstruction of PDE solutions in Bui-Thanh *et al.* (2003, 2004). The gappy POD has also served as the foundation for the nonlinear model reduction methods of missing point estimation (Astrid *et al.*, 2008) and Gauss Newton with approximated tensors (GNAT) (Carlberg *et al.*, 2013).

We define a mask vector, which describes for a particular state vector where data are available and where data are missing. For the state snapshot $\boldsymbol{u}$, the corresponding mask vector $\boldsymbol{n}$ is defined as

$$n_i = 0, \text{ if } u_i \text{ is missing}$$
$$n_i = 1, \text{ if } u_i \text{ is known}$$

where u_i and n_i denote the ith element of the vectors $\boldsymbol{u}$ and $\boldsymbol{n}$, respectively. Pointwise multiplication is defined as $(\boldsymbol{n},\boldsymbol{u})_i = n_i u_i$. The gappy inner product is defined as $(\boldsymbol{u},\boldsymbol{v})_n = ((\boldsymbol{n},\boldsymbol{u}),(\boldsymbol{n},\boldsymbol{v}))$, and the induced norm is $(\|\boldsymbol{v}\|_n)^2 = (\boldsymbol{v},\boldsymbol{v})_n$.

Let $\{\boldsymbol{v}_k\}_{k=1}^r$ be the standard POD basis for the snapshot set $\{\boldsymbol{u}_k\}_{k=1}^m$, where all snapshots are completely known, and the POD basis vectors $\boldsymbol{v}$ are computed as described in Section 2.2. Let $\boldsymbol{g}$ be another state vector that has some elements missing, with corresponding mask vector $\boldsymbol{n}$. We wish to reconstruct the full or "repaired" vector from the incomplete vector $\boldsymbol{g}$. This setting might arise, for example, when $\boldsymbol{g}$ represents sensor data that provide sparse measurements of the system state (e.g., pressure sensors on an aircraft wing), and there is a desire to reconstruct the corresponding full state information.

The gappy POD defines the intermediate repaired vector $\tilde{\boldsymbol{g}}$ to be a linear combination of r POD basis vectors

$$\tilde{\boldsymbol{g}} = \sum_{i=1}^{r} b_i \boldsymbol{v}_i \tag{18}$$

To compute the POD coefficients b_i, we minimize the error ϵ between the original and repaired vectors, computed over those elements for which data are available:

$$\epsilon = \|\boldsymbol{g} - \tilde{\boldsymbol{g}}\|_n^2 \tag{19}$$

The coefficients b_i that minimize the error ϵ can be found by using the definition (18) and differentiating (19) with respect to each of the b_i in turn. This leads to the linear system of equations

$$\boldsymbol{M}\boldsymbol{b} = \boldsymbol{d} \tag{20}$$

where

$$M_{ij} = (\boldsymbol{v}_i, \boldsymbol{v}_j)_n \tag{21}$$

and

$$d_i = (\boldsymbol{g}, \boldsymbol{v}_i)_n \tag{22}$$

Solving equation (20) for $\boldsymbol{b}$ and using (18) defines the intermediate repaired vector $\tilde{\boldsymbol{g}}$. Finally, the complete $\boldsymbol{g}$ is reconstructed by replacing the missing elements in $\boldsymbol{g}$ by the corresponding repaired elements in $\tilde{\boldsymbol{g}}$, that is, set $g_i = \tilde{g}_i$ if $n_i = 0$.

While not discussed here, we also note that if the original snapshot ensemble has incomplete data, the POD basis vectors can be computed using an iterative gappy approach, see Everson and Sirovich (1995) and Bui-Thanh *et al.* (2004).

2.6 Summary

POD is a powerful model reduction method, because of its versatility and the wide range of applications for which it has proven effective. For example, the POD method of snapshots has been used widely throughout computational fluid dynamics applications such as aeroelasticity (see e.g., Dowell and Hall, 2001; Lieu *et al.*, 2006) and flow control (see e.g., Kunisch and Volkwein, 1999; Hinze and Volkwein, 2005). Increases in sensor data availability in systems, combined with increased on-board computing power, offer new opportunities to dynamically adapt reduced models and to increase their usefulness in real-time decision-making situations. Early work in dynamic data-driven adaptation of POD models shows promise and is a productive area to explore further. As the application of POD model reduction to more complicated systems becomes more widespread, the question of a priori guarantees of reduced model accuracy and robustness remains an important open challenge.

3 REDUCED BASIS METHODS

3.1 Introduction

We now consider (hierarchical, Lagrange) reduced basis (RB) approximation and *a posteriori* error estimation for linear functional outputs of affinely parametrized elliptic coercive PDEs. These assumptions do not limit the applicability of the methodology to a broader class of problems. The essential ingredients are as follows: a Galerkin projection onto a low-dimensional space – dimension reduction; efficient and effective greedy sampling methods for identification of optimal and numerically stable approximations – rapid convergence; *a posteriori* error estimation procedures – rigorous and sharp bounds for the linear-functional outputs of interest; and offline–online computational decomposition strategies – minimum *marginal cost* for high performance in the real-time/embedded (e.g., parameter estimation, control) and many-query (e.g., design optimization, multimodel/scale) contexts. In this overview, we describe the basic ideas of reduced basis approximation methods for rapid and reliable evaluation of *input–output relationships* in which the *output* is expressed as a functional of a *field variable* that is the solution of an *input-parametrized* PDE. The combination with an efficient *a posteriori* error estimation is a key factor for RB methods to be computationally successful.

Parametrized PDEs model several processes that are relevant in applications, such as, for example, unsteady and steady heat and mass transfer, acoustics, and solid and fluid mechanics, but also electromagnetics or even finance. The *input-parameter* vector may characterize either the geometric configuration, some physical properties, or boundary conditions and source terms. The *outputs of interest* might be the maximum system temperature, an added mass coefficient, a crack stress intensity factor, an effective constitutive property, an acoustic waveguide transmission loss, or a channel flowrate or pressure drop, just to mention a few. Finally, the *field variables* that connect the input parameters to the outputs can represent a distribution function, temperature or concentration, displacement, pressure, or velocity.

The class of problems we consider – the case of linear functional outputs of affinely parametrized linear elliptic coercive PDEs – is relatively simple, yet relevant to many important applications in transport (e.g., conduction and convection-diffusion) and continuum mechanics (e.g., linear elasticity) and proves a convenient expository vehicle for the methodology.

Furthermore, most of the basic concepts introduced in the affine linear elliptic coercive case are equally crucial – with suitable extension – to more general reduced basis approximation and *a posteriori* error estimation methodology, see for example, reviews already mentioned: Prud'homme *et al.* (2002), Rozza *et al.* (2008), Patera and Rozza (2007), Quarteroni *et al.* (2011), Rozza (2014), and Hesthaven *et al.* (2015).

The RB methodology we describe here is motivated by, optimized for, and applied within two particular contexts: the *real-time context* (e.g., parameter estimation or control), and the *many-query context* (e.g., design optimization or multimodel/scale simulation). Both are crucial to computational engineering. We note, however, that the RB methods we describe do not replace, but rather build upon and are measured (as regards accuracy) relative to, a classical discretization technique (finite element, finite volume, spectral methods, etc.): the reduced basis approximates not the exact solution but rather a "given" finite discretization of (typically) very large dimension N_h. In short, we pursue an

algorithmic collaboration rather than an algorithmic competition with the classical finite discretization methods.

We provide here a brief outline. In Section 3.2 we describe the affine linear elliptic coercive setting; in Section 3.3 we consider classes of piecewise-affine geometry and coefficient parametric variation; Section 3.4 deals with parametric formulation for bilinear and linear forms. Section 3.5 is strictly devoted to reduced basis methodology, in particular in Section 3.5.1 we discuss RB Galerkin projection and optimality; in Section 3.6 we describe greedy sampling procedures for optimal space identification; in Section 3.7 we briefly recall the *a priori* convergence theory of the methodology. In Section 3.8 we present rigorous and relatively sharp *a posteriori* output error bounds for RB approximations but without developing here the coercivity-constant lower bounds required by the *a posteriori* error estimation procedures. In Section 3.9 we make few comments on historical background, current state of the art, and future perspectives of developments for the RB methodology.

3.2 Linear elliptic coercive affine parametric PDEs

In an abstract form, we consider the following problem: Given $\boldsymbol{\mu} \in \mathcal{D} \subset \mathbb{R}^P$, for an integer $P \geq 1$, evaluate

$$s^{\mathrm{e}}(\boldsymbol{\mu}) = L(u^{\mathrm{e}}(\boldsymbol{\mu})) \tag{23}$$

where $u^{\mathrm{e}}(\boldsymbol{\mu}) \in V^{\mathrm{e}}(\Omega)$ satisfies

$$a(u^{\mathrm{e}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = F(v) \quad \forall\, v \in V^{\mathrm{e}} \tag{24}$$

The superscript $^{\mathrm{e}}$ refers to "exact". Here, $\boldsymbol{\mu}$ is the input parameter – a P-tuple; $\mathcal{D}$ is the parameter domain – a subset of $\mathbb{R}^P$; s^{e} is the scalar output; L is the linear output functional, that is, a linear and bounded functional on V^{e}; u^{e} is the field variable; Ω is the bounded spatial domain in $\mathbb{R}^d$ (for $d = 2$ or 3) with Lipschitz boundary $\partial\Omega$; V^{e} is a Hilbert space; and a and F are the bilinear and linear forms, respectively, associated with our PDE.

We shall exclusively consider second-order elliptic PDEs, in which case $V^{\mathrm{e}} = H^1_{\Gamma_D}(\Omega)$ is the subspace of $H^1_0(\Omega)$, the one of functions that vanish on the part of $\partial\Omega$ where Dirichlet boundary data are prescribed for u^{e}.

We shall assume that the bilinear form $a(\cdot, \cdot; \boldsymbol{\mu})$: $V^{\mathrm{e}} \times V^{\mathrm{e}} \to \mathbb{R}$ is continuous (with continuity constant $\gamma^{\mathrm{e}}(\boldsymbol{\mu})$) and coercive (with coercivity constant $\alpha^{\mathrm{e}}(\boldsymbol{\mu})$) over V^{e} for all $\boldsymbol{\mu}$ in $\mathcal{D}$. We further assume that F is a bounded linear functional over V^{e}. Under these standard hypotheses on a and F, (24) admits a unique solution, thanks to the Lax–Milgram lemma.

We assume that (i) a is symmetric, and furthermore (ii) $L = F$. The last assumption is made for simplification, and it means that we are in the so-called *compliant* case (Rozza *et al.*, 2008), a situation occurring quite frequently in engineering problems. Extension to the noncompliant case in which now a may be nonsymmetric and L may be any bounded linear functional over V^{e} is described in Hesthaven *et al.* (2015), for example.

We shall make one last assumption, crucial to the enhancement of computational efficiency: the parametric bilinear form a is *affine* w.r.t. the parameter $\boldsymbol{\mu}$, by which we mean

$$a(w, v; \boldsymbol{\mu}) = \sum_{q=1}^{Q} \Theta^q(\boldsymbol{\mu})\, a^q(w, v) \quad \forall v, w \in V^{\mathrm{e}}, \boldsymbol{\mu} \in \mathcal{D} \tag{25}$$

Here, for $q = 1, \ldots, Q$, $\Theta^q : \mathcal{D} \to \mathbb{R}$ is a $\boldsymbol{\mu}$-dependent function, whereas $a^q : V^{\mathrm{e}} \times V^{\mathrm{e}} \to \mathbb{R}$ is $\boldsymbol{\mu}$-independent. In actual practice, F may also depend affinely on the parameter: in this case, $F(v; \boldsymbol{\mu})$ may be expressed as a sum of Q^f products of parameter-dependent functions and parameter-independent bounded linear functionals on V^{e}.

We next proceed to the finite-dimensional approximation of problem (24) by any kind of Galerkin method, for instance, the finite element (FE) method: Given $\boldsymbol{\mu} \in \mathcal{D} \subset \mathbb{R}^P$, evaluate

$$s^{N_h}(\boldsymbol{\mu}) = F(u^{N_h}(\boldsymbol{\mu}))$$

(recall our compliance assumption: $L = F$), where $u^{N_h}(\boldsymbol{\mu}) \in V^{N_h} \subset V^{\mathrm{e}}$ satisfies

$$a(u^{N_h}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = F(v) \quad \forall\, v \in V^{N_h} \tag{26}$$

Here, $V^{N_h} \subset V^{\mathrm{e}}$ is a sequence of FE approximation spaces indexed by $\dim(V^{N_h}) = N_h$. It follows directly from our assumptions on a, F, and V^{N_h} that (26) admits a unique solution. Our RB field and RB output shall approximate, for given N_h, the FE solution $u^{N_h}(\boldsymbol{\mu})$ and FE output $s^{N_h}(\boldsymbol{\mu})$ (hence, indirectly, $u^{\mathrm{e}}(\boldsymbol{\mu})$ and $s^{\mathrm{e}}(\boldsymbol{\mu})$).

We can now define the energy inner product and the energy norm for elements of V^{e}:

$$(w, v)_{\boldsymbol{\mu}} = a(w, v; \boldsymbol{\mu}) \quad \forall\, w, v \in V^{\mathrm{e}}\ , \tag{27}$$

$$\|w\|_{\boldsymbol{\mu}} = (w, w)^{1/2}_{\boldsymbol{\mu}} \quad \forall\, w \in V^{\mathrm{e}} \tag{28}$$

Next, for given $\overline{\boldsymbol{\mu}} \in \mathcal{D}$ and nonnegative real τ

$$(w, v)_V = (w, v)_{\overline{\boldsymbol{\mu}}} + \tau (w, v)_{L^2(\Omega)} \quad \forall\, w, v \in V^{\mathrm{e}} \tag{29}$$

$$\|w\|_V = (w, w)^{1/2}_V \qquad \forall\, w, v \in V^{\mathrm{e}}$$

shall define our V inner product and norm, respectively.

Finally, we can now define more precisely our coercivity and continuity constants (and coercivity and continuity

conditions). In particular, we define the exact and FE coercivity constants as

$$\alpha^{\mathrm{e}}(\boldsymbol{\mu}) = \inf_{w \in V^{\mathrm{e}}} \frac{a(w, w; \boldsymbol{\mu})}{\|w\|_V^2} \tag{30}$$

and

$$\alpha^{N_h}(\boldsymbol{\mu}) = \inf_{w \in V^{N_h}} \frac{a(w, w; \boldsymbol{\mu})}{\|w\|_V^2} \tag{31}$$

respectively. It follows from the coercivity hypothesis that there exists a positive constant α_0 s.t. $\alpha^{\mathrm{e}}(\boldsymbol{\mu}) \geq \alpha_0 > 0 \ \forall \boldsymbol{\mu} \in \mathcal{D}$, and that $\alpha^{N_h}(\boldsymbol{\mu}) \geq \alpha^{\mathrm{e}}(\boldsymbol{\mu}) \, \forall \boldsymbol{\mu} \in \mathcal{D}$. Similarly, we define the exact and FE continuity constants as

$$\gamma^{\mathrm{e}}(\boldsymbol{\mu}) = \sup_{w \in V^{\mathrm{e}}} \sup_{v \in V^{\mathrm{e}}} \frac{a(w, v; \boldsymbol{\mu})}{\|w\|_V \|v\|_V} \tag{32}$$

and

$$\gamma^{N_h}(\boldsymbol{\mu}) = \sup_{w \in V^{N_h}} \sup_{v \in V^{N_h}} \frac{a(w, v; \boldsymbol{\mu})}{\|w\|_V \|v\|_V} \tag{33}$$

respectively. It follows from the continuity hypothesis on a that $\gamma^{\mathrm{e}}(\boldsymbol{\mu})$ is finite $\forall \boldsymbol{\mu} \in \mathcal{D}$, and that $\gamma^{N_h}(\boldsymbol{\mu}) \leq \gamma^{\mathrm{e}}(\boldsymbol{\mu}) \ \forall \boldsymbol{\mu} \in \mathcal{D}$.

3.3 Geometric parametrization

We introduce the family of geometric parametric variations consistent with our affine restriction (25) as a general class of scalar problems that undergo the abstract formulation of Section 3.2. For simplicity, we consider only the scalar case; the vector case permits an analogous treatment.

3.3.1 Affine geometry precondition

The RB method requires that Ω be a *parameter-independent* domain: if we wish to consider linear combinations of precomputed solutions (also called "snapshots"), the latter must be defined relative to a common spatial configuration. Thus, to permit geometric variation we must interpret Ω, our parameter-independent reference domain, as the pre-image of $\Omega_{\mathrm{o}}(\boldsymbol{\mu})$, the parameter-dependent "actual" or "original" domain of interest. The geometric transformation will yield variable (parameter-dependent) coefficients of linear and bilinear forms in the reference domain, which, under suitable hypotheses to be discussed below, will take the requisite affine form (25).

We shall assume that, for all $\boldsymbol{\mu}$ in $\mathcal{D}$, $\Omega_{\mathrm{o}}(\boldsymbol{\mu})$ is expressed as a domain decomposition such that

$$\overline{\Omega}_{\mathrm{o}}(\boldsymbol{\mu}) = \cup_{k=1}^{K_{\mathrm{dom}}} \overline{\Omega}_{\mathrm{o}}^k(\boldsymbol{\mu}) \tag{34}$$

where the $\Omega_{\mathrm{o}}^k(\boldsymbol{\mu}), 1 \leq k \leq K_{\mathrm{dom}}$, are mutually nonoverlapping open subdomains.[2]

$$\Omega_{\mathrm{o}}^k(\boldsymbol{\mu}) \cap \Omega_{\mathrm{o}}^{k'}(\boldsymbol{\mu}) = \emptyset, \quad 1 \leq k < k' \leq K_{\mathrm{dom}} \tag{35}$$

We now choose a value $\boldsymbol{\mu}_{\mathrm{ref}} \in \mathcal{D}$ and define our reference domain as $\Omega = \Omega_{\mathrm{o}}(\boldsymbol{\mu}_{\mathrm{ref}})$. It immediately follows from (34) and (35) that

$$\overline{\Omega} = \cup_{k=1}^{K_{\mathrm{dom}}} \overline{\Omega}^k \tag{36}$$

$$\Omega^k \cap \Omega^{k'} = \emptyset, \quad 1 \leq k < k' \leq K_{\mathrm{dom}} \tag{37}$$

for $\Omega^k = \Omega_{\mathrm{o}}^k(\boldsymbol{\mu}_{\mathrm{ref}}), 1 \leq k \leq K_{\mathrm{dom}}$.

We will build our FE approximation on a very fine FE subtriangulation of the coarse domain decomposition partition. The latter can be called, for simplicity, the RB triangulation of Ω. (Recall that both the FE and RB approximations are defined over the reference domain.) Note that we purposely define K_{dom} with respect to the *exact* problem, rather than the FE approximation: K_{dom} cannot depend on the FE subgrid to be meaningful. This FE subtriangulation ensures that the FE approximation accurately treats the perhaps discontinuous coefficients (arising from property and geometry variation) associated with the different regions; the subtriangulation also plays an important role in the generation of the affine representation (25).

We emphasize that the choice of $\boldsymbol{\mu}_{\mathrm{ref}}$ only affects the accuracy of the underlying FE approximation upon which the RB discretization and *a posteriori* error estimator is built: typically, a value of $\boldsymbol{\mu}_{\mathrm{ref}}$ at the "center" of $\mathcal{D}$ minimizes distortion and reduces the size N_h of the finite element problem necessary to yield a given acceptable FE error over $\mathcal{D}$.

We can treat any original domain $\Omega_{\mathrm{o}}(\boldsymbol{\mu})$ that admits a domain partition (34)–(35) for which $\forall \boldsymbol{\mu} \in \mathcal{D}$

$$\overline{\Omega}_{\mathrm{o}}^k(\boldsymbol{\mu}) = \mathcal{T}^k(\overline{\Omega}^k; \boldsymbol{\mu}), \quad 1 \leq k \leq K_{\mathrm{dom}} \tag{38}$$

for *affine* mappings $\mathcal{T}^k(\cdot; \boldsymbol{\mu})$: $\Omega^k \to \Omega_{\mathrm{o}}^k(\boldsymbol{\mu})$, $1 \leq k \leq K_{\mathrm{dom}}$, that are

(i) individually *bijective* (they induce the same subdivision from either side of an interface), and
(ii) collectively *continuous*

$$\mathcal{T}^k(\boldsymbol{x}; \boldsymbol{\mu}) = \mathcal{T}^{k'}(\boldsymbol{x}; \boldsymbol{\mu}) \quad \forall \boldsymbol{x} \in \overline{\Omega}^k \cap \overline{\Omega}^{k'}, 1 \leq k < k' \leq K_{\mathrm{dom}} \tag{39}$$

The affine geometry precondition is a necessary condition for affine parameter dependence as defined in (25).

We now define the bijective affine mappings more explicitly: for $1 \le k \le K_{\text{dom}}$, any $\boldsymbol{\mu}$ in $\mathcal{D}$, and any $\boldsymbol{x} \in \Omega^k$

$$\mathcal{T}_i^k(\boldsymbol{x};\boldsymbol{\mu}) = C_i^k(\boldsymbol{\mu}) + \sum_{j=1}^{d} G_{i\,j}^k(\boldsymbol{\mu})\, x_j, \quad 1 \le i \le d \tag{40}$$

for given $C^k: \mathcal{D} \to \mathbb{R}^d$ and $G^k: \mathcal{D} \to \mathbb{R}^{d\times d}$. We can then define the associated Jacobians

$$J^k(\boldsymbol{\mu}) = |\det(G^k(\boldsymbol{\mu}))|, \quad 1 \le k \le K_{\text{dom}} \tag{41}$$

where det denotes the determinant; note the Jacobian is constant in space over each subdomain. We further define, for any $\boldsymbol{\mu} \in \mathcal{D}$

$$D^k(\boldsymbol{\mu}) = (G^k(\boldsymbol{\mu}))^{-1}, \quad 1 \le k \le K_{\text{dom}} \tag{42}$$

This matrix shall prove convenient in subsequent transformations involving derivatives.

We may interpret our local mappings in terms of a global transformation. In particular, for any $\boldsymbol{\mu} \in \mathcal{D}$, the local mappings (38) induce a global bijective *piecewise-affine* transformation $\mathcal{T}(\,\cdot\,;\boldsymbol{\mu})$: $\Omega \to \Omega_{\text{o}}(\boldsymbol{\mu})$: for any $\boldsymbol{\mu} \in \mathcal{D}$

$$\mathcal{T}(\boldsymbol{x};\boldsymbol{\mu}) = \mathcal{T}^k(\boldsymbol{x};\boldsymbol{\mu}), \quad k = \min_{\{k' \in \{1,\dots,K_{\text{dom}}\} \mid x \in \overline{\Omega}^{k'}\}} k' \tag{43}$$

Note the one-to-one property of this mapping (and, hence the arbitrariness of our "min" choice in (43)) is ensured by the interface condition (39). We can further demonstrate that these global *continuous* mappings are compatible with our second-order PDE variational formulation: for any $\boldsymbol{\mu} \in \mathcal{D}$, given any $w_{\text{o}} \in H^1(\Omega_{\text{o}}(\boldsymbol{\mu}))$, $w = w_{\text{o}} \circ \mathcal{T} \in H^1(\Omega)$; this ensures that our mapped problem on the reference domain is of the classical conforming type.

Although this concludes a very basic formal exposition of admissible geometry variations, the application of these conditions requires familiarity with the scope of the affine mappings (40). In Section 3.4 we discuss the incorporation of these affine mappings into our weak form.

3.4 Parametric bilinear forms

As already indicated, we shall consider here only the scalar case; the vector case, for instance, arising from linear elasticity, allows an analogous treatment.

3.4.1 Formulation on "original" domain

Our problem is initially posed on the "original" domain $\Omega_{\text{o}}(\boldsymbol{\mu})$, which we assume realizes the affine geometry precondition as described in the previous section. We shall assume for simplicity that $V_{\text{o}}^{\text{e}}(\boldsymbol{\mu}) = H_0^1(\Omega_{\text{o}}(\boldsymbol{\mu}))$, which corresponds to homogeneous Dirichlet boundary conditions over the entire boundary $\partial\Omega_{\text{o}}(\boldsymbol{\mu})$; we subsequently discuss natural (Neumann and Robin) conditions.

Given $\boldsymbol{\mu} \in \mathcal{D}$, we evaluate

$$s_{\text{o}}^{\text{e}}(\boldsymbol{\mu}) = F_{\text{o}}(u_{\text{o}}^{\text{e}}(\boldsymbol{\mu}))$$

where $u_{\text{o}}^{\text{e}}(\boldsymbol{\mu}) \in V_{\text{o}}^{\text{e}}(\boldsymbol{\mu})$ satisfies

$$a_{\text{o}}(u_{\text{o}}^{\text{e}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = F_{\text{o}}(v) \quad \forall v \in V_{\text{o}}^{\text{e}}(\boldsymbol{\mu})$$

We now place conditions on a_{o} and F_{o} such that, in conjunction with the affine geometry precondition, we are ensured an affine expansion of the bilinear form.

In particular, we require $a_{\text{o}}(\cdot,\cdot;\boldsymbol{\mu})$: $H^1(\Omega_{\text{o}}(\boldsymbol{\mu})) \times H^1(\Omega_{\text{o}}(\boldsymbol{\mu})) \to \mathbb{R}$ to be expressed as

$$a_{\text{o}}(w, v; \boldsymbol{\mu}) = \sum_{\ell=1}^{K_{\text{dom}}} \int_{\Omega_{\text{o}}^\ell(\boldsymbol{\mu})} \begin{bmatrix} \dfrac{\partial w}{\partial x_{\text{o}1}} & \dfrac{\partial w}{\partial x_{\text{o}2}} & w \end{bmatrix} \mathcal{K}_{\text{o},\ell\;ij}(\boldsymbol{\mu}) \begin{bmatrix} \dfrac{\partial v}{\partial x_{\text{o}1}} \\ \dfrac{\partial v}{\partial x_{\text{o}2}} \\ v \end{bmatrix} \mathrm{d}\Omega_{\text{o}} \tag{44}$$

where $\boldsymbol{x}_{\text{o}} = (x_{\text{o}1}, x_{\text{o}2})$ denotes a point in $\Omega_{\text{o}}(\boldsymbol{\mu})$. Here, for $1 \le \ell \le K_{\text{dom}}$, $\mathcal{K}_{\text{o},\ell}: \mathcal{D} \to \mathbb{R}^{3\times 3}$ is a given symmetric positive definite matrix, which in turn ensures coercivity of our bilinear form; the upper 2×2 principal submatrix of $\mathcal{K}_{\text{o},\ell}$ is the usual tensor conductivity/diffusivity; the $(3,3)$ element of $\mathcal{K}_{\text{o},\ell}$ represents the identity operator, leading to the mass matrix. The $(3,1), (3,2)$ (and $(1,3), (2,3)$) elements of $\mathcal{K}_{\text{o},\ell}$ permit first derivative (or convective) terms.

Similarly, we require that F_{o}: $H^1(\Omega_{\text{o}}(\boldsymbol{\mu})) \to \mathbb{R}$ can be expressed as

$$F_{\text{o}}(v) = \sum_{\ell=1}^{K_{\text{dom}}} \int_{\Omega_{\text{o}}^\ell(\boldsymbol{\mu})} \mathcal{F}_{\text{o},\ell}(\boldsymbol{\mu}) v \, \mathrm{d}\Omega_{\text{o}}$$

where, for $1 \le \ell \le K_{\text{dom}}$, $\mathcal{F}_{\text{o},\ell}: \mathcal{D} \to \mathbb{R}$.

3.4.2 Formulation on reference domain

We now apply standard techniques to transform the problem statement over the original domain to an equivalent problem statement over the reference domain: given $\boldsymbol{\mu} \in \mathcal{D}$, we find

$$s^{\text{e}}(\boldsymbol{\mu}) = F(u^{\text{e}}(\boldsymbol{\mu}))$$

where $u^{\text{e}}(\boldsymbol{\mu}) \in V^{\text{e}} = H_0^1(\Omega)$ satisfies

$$a(u^{\text{e}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = F(v) \quad \forall v \in V^{\text{e}}$$

We may then identify $s^e(\boldsymbol{\mu}) = s^e_o(\boldsymbol{\mu})$ and $u^e(\boldsymbol{\mu}) = u^e_o\ (\boldsymbol{\mu}) \cdot \mathcal{T}(\cdot;\boldsymbol{\mu})$.

The transformed bilinear form a can be expressed as

$$a(w,v;\boldsymbol{\mu}) = \sum_{k=1}^{K_{\text{dom}}} \int_{\Omega^k} \begin{bmatrix} \dfrac{\partial w}{\partial x_1} & \dfrac{\partial w}{\partial x_2} & w \end{bmatrix} \mathcal{K}^k_{ij}(\boldsymbol{\mu}) \begin{bmatrix} \dfrac{\partial v}{\partial x_1} \\ \dfrac{\partial v}{\partial x_2} \\ v \end{bmatrix} \mathrm{d}\Omega \tag{45}$$

where $\boldsymbol{x} = (x_1, x_2)$ denotes a point in Ω. Here, $\mathcal{K}^k : \mathcal{D} \to \mathbb{R}^{3\times 3}$, $1 \le k \le K_{\text{dom}}$, are symmetric positive-definite matrices given by

$$\mathcal{K}^k(\boldsymbol{\mu}) = J^k(\boldsymbol{\mu})\, \mathcal{G}^k(\boldsymbol{\mu})\, \mathcal{K}_{o,\ell}(\boldsymbol{\mu})(\mathcal{G}^k(\boldsymbol{\mu}))^{\mathrm{T}} \tag{46}$$

for $1 \le k \le K_{\text{dom}}$; the $\mathcal{G}^k : \mathcal{D} \to \mathbb{R}^{3\times 3}$, $1 \le k \le K_{\text{dom}}$, are given by

$$\mathcal{G}^k(\boldsymbol{\mu}) = \begin{pmatrix} D^k(\boldsymbol{\mu}) & 0 \\ 0\ \ 0 & 0 \\ 0\ \ 0 & 1 \end{pmatrix} \tag{47}$$

$J^k(\boldsymbol{\mu})$ and $D^k(\boldsymbol{\mu})$, $1 \le k \le K_{\text{dom}}$, are given by (41) and (42), respectively.

Similarly, the transformed linear form can be expressed as

$$F(v) = \sum_{k=1}^{K_{\text{dom}}} \int_{\Omega^k} \mathcal{F}^k(\boldsymbol{\mu}) v \ \mathrm{d}\Omega$$

Here, $\mathcal{F}^k : \mathcal{D} \to \mathbb{R}$, $1 \le k \le K_{\text{dom}}$, is given by

$$\mathcal{F}^k = J^k(\boldsymbol{\mu}) \mathcal{F}_{o,\ell}(\boldsymbol{\mu}), \quad 1 \le k \le K_{\text{dom}}$$

We note that, in general, $\mathcal{K}^k(\boldsymbol{\mu})$ and $\mathcal{F}^k(\boldsymbol{\mu})$, $1 \le k \le K_{\text{dom}}$, will be different for each subdomain Ω^k. The differences can arise either because of coefficient variation, or because of geometry variation, or both. We thus require, as already indicated earlier, that the FE approximation be built upon a subtriangulation of the RB triangulation. Discontinuities in PDE coefficients are thereby restricted to element edges to ensure rapid convergence; and identification/extraction of the terms in the affine expansion (25) is more readily effected, as we now discuss.

3.4.3 Affine forms

We focus here on a, though F admits a similar treatment. We simply expand the form (45) by considering in turn each subdomain Ω^k and each entry of the diffusivity tensor $\mathcal{K}^k_{ij}$, $1 \le i,j \le 3$, $1 \le k \le K_{\text{dom}}$. Thus

$$\begin{aligned} a(w,v;\boldsymbol{\mu}) &= \mathcal{K}^1_{11}(\boldsymbol{\mu}) \int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_1} \mathrm{d}\Omega \\ &+ \mathcal{K}^1_{12}(\boldsymbol{\mu}) \int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_2} d\Omega + \cdots + \mathcal{K}^{K_{\text{dom}}}_{33}(\boldsymbol{\mu}) \int_{\Omega^{K_{\text{dom}}}} wv \ \mathrm{d}\Omega \end{aligned} \tag{48}$$

We can then identify each component in the affine expansion: for each term in (48), the prefactor represents $\Theta^q(\boldsymbol{\mu})$, while the integral represents a^q.

Taking into account the symmetry of the bilinear form, such that only the $(1,1)$, $(1,2)$ $(= (2,1))$, $(2,2)$, and $(3,3)$ entries of $\mathcal{K}_{o,\ell}(\boldsymbol{\mu})$ – and hence $\mathcal{K}^k(\boldsymbol{\mu})$ – must be accommodated, there are at most $Q = 4K$ terms in the affine expansion. $\Theta^q(\boldsymbol{\mu})$ are given by, for the obvious numbering scheme $\Theta^1(\boldsymbol{\mu}) = \mathcal{K}^1_{11}(\boldsymbol{\mu}), \Theta^2(\boldsymbol{\mu}) = \mathcal{K}^1_{12}(\boldsymbol{\mu}), \ldots, \Theta^5(\boldsymbol{\mu}) = \mathcal{K}^2_{11}(\boldsymbol{\mu}), \ldots, \Theta^Q(\boldsymbol{\mu}) = \mathcal{K}^{K_{\text{dom}}}_{33}(\boldsymbol{\mu})$; $a^q(w,v)$ are given by

$$\begin{aligned} a^1(w,v) &= \int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_1} \mathrm{d}\Omega \\ a^2(w,v) &= \int_{\Omega^1} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_2} \mathrm{d}\Omega \\ &\vdots \\ a^5(w,v) &= \int_{\Omega^2} \frac{\partial w}{\partial x_1} \frac{\partial v}{\partial x_1} \mathrm{d}\Omega \\ &\vdots \\ a^Q(w,v) &= \int_{\Omega^{K_{\text{dom}}}} wv \ \mathrm{d}\Omega \end{aligned}$$

This identification constitutes a constructive proof that the affine geometry precondition and the property/coefficient variation permitted by (44) do indeed yield a bilinear form which can be expressed in the requisite affine form (25).

We close with a discussion of generality. In fact, the conditions we provide are sufficient but not necessary. For example, we can permit affine polynomial dependence on $\boldsymbol{x}_o$ in both $\mathcal{K}_{o,k}(\boldsymbol{x}_o;\boldsymbol{\mu})$ and $\mathcal{F}_{o,k}(\boldsymbol{x}_o;\boldsymbol{\mu})$ and still ensure an affine development, (25); furthermore, in the absence of geometric variation, $\mathcal{K}_{o,\ell}(\boldsymbol{x}_o;\boldsymbol{\mu})$ and $\mathcal{F}_{o,\ell}(\boldsymbol{x}_o;\boldsymbol{\mu})$ can take on *any* "separable" form in $\boldsymbol{x}, \boldsymbol{\mu}$. However, the affine expansion (25) is by no means completely general: for more complicated parametric dependencies, non-affine techniques held by the empirical interpolation methods family (Barrault *et al.*, 2004; Grepl *et al.*, 2007; Rozza, 2009) must be invoked. In Hesthaven *et al.* (2015), several variations of empirical interpolation methods are recalled and compared.

3.5 The reduced basis method

We are eventually ready to illustrate the essence of the RB methodology.

3.5.1 *Reduced basis approximation and spaces*

We assume that we are given an FE approximation space V^{N_h} of dimension N_h. In order to define a particular reduced basis space, we start by considering a fixed N_h. We then introduce, given a positive integer $N_{\max}$, an associated sequence of what shall ultimately be the RB approximation spaces: for $N = 1, \ldots, N_{\max}$, $X_N^{N_h}$ is an N-dimensional subspace of V^{N_h}; we further suppose that

$$V_1^{N_h} \subset V_2^{N_h} \subset \cdots V_{N_{\max}}^{N_h} \subset V^{N_h} \tag{49}$$

As we shall see, the nested or *hierarchical* condition (49) is important in ensuring the memory efficiency of the resulting RB approximation. As we will mention in Section 3.9, there are several classical RB proposals – Taylor, Lagrange (Porsching, 1985), and Hermite (Ito and Ravindran, 1998) spaces – as well as several more recent alternatives – such as POD spaces, as introduced in Section 2. All of these spaces "focus" in one fashion or another on a low-dimensional, smooth parametric manifold, $\mathcal{M}^{N_h} = \{u(\boldsymbol{\mu}) \mid \boldsymbol{\mu} \in \mathcal{D}\}$: the set of fields engendered as the input varies over the parameter domain. In the case of a single parameter, the parametrically induced manifold is a one-dimensional filament within the infinite dimensional space which characterizes *general* solutions to the given PDE. Clearly, generic approximation spaces are unnecessarily rich and hence are unnecessarily expensive within the parametric framework. Much of what we present – in particular, all the discussion of this section related to optimality, discrete equations, conditioning, and offline–online procedures, and all that of Section 3.8 related to *a posteriori* error estimation – shall be relevant to any of these reduced basis spaces/approximations.

However, some of what we shall present, in particular related to sampling strategies in Section 3.6, is restricted to the particular reduced basis space, which shall be our primary focus: the Lagrange reduced basis spaces (Porsching, 1985), which we denote by $W_N^{N_h}$. In order to define a (hierarchical) sequence of Lagrange spaces $W_N^{N_h}$, $1 \le N \le N_{\max}$, we first introduce a master set of parameter points $\boldsymbol{\mu}^n \in \mathcal{D}$, $1 \le n \le N_{\max}$. We then define, for a given $N \in \{1, \ldots, N_{\max}\}$, the Lagrange parameter samples

$$S_N = \{\boldsymbol{\mu}^1, \ldots, \boldsymbol{\mu}^N\} \tag{50}$$

and the associated Lagrange RB spaces

$$W_N^{N_h} = \text{span}\{u^{N_h}(\boldsymbol{\mu}^n), \quad 1 \le n \le N\} \tag{51}$$

We observe that, by construction, these Lagrange spaces $W_N^{N_h}$ satisfy (49): the samples (50) are nested, that is $S_1 = \{\boldsymbol{\mu}^1\} \subset S_2 = \{\boldsymbol{\mu}^1, \boldsymbol{\mu}^2\} \subset \cdots \subset S_{N_{\max}}$; the Lagrange RB spaces (51) are hierarchical, that is, $W_1^{N_h} = \text{span}\{u^{N_h}(\boldsymbol{\mu}^1)\} \subset W_2^{N_h} = \text{span}\{u^{N_h}(\boldsymbol{\mu}^1), u^{N_h}(\boldsymbol{\mu}^2)\} \subset \cdots \subset W_{N_{\max}}^{N_h}$.

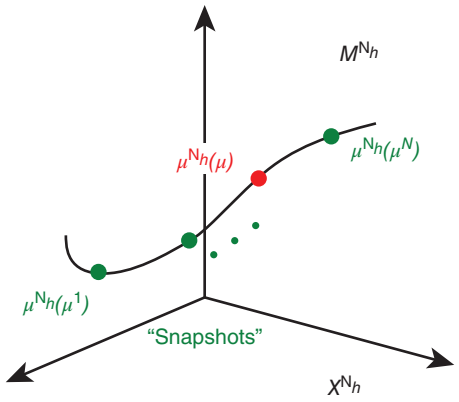

Figure 1. "Snapshots" $u^{N_h}(\boldsymbol{\mu}^n)$, $1 \le n \le N$, on the parametric manifold $\mathcal{M}^{N_h}$.

The $u^{N_h}(\boldsymbol{\mu}^n)$, $1 \le n \le N_{\max}$, are often referred to as *snapshots* of the parametric manifold $\mathcal{M}^{N_h}$. For reasons that will become clear subsequently, these snapshots are more precisely referred to as *retained* snapshots. We depict the retained snapshots graphically in Figure 1. It is clear that, if indeed the manifold is low-dimensional and smooth (a point we will return to later), then we would expect to well approximate any member of the manifold – any solution $u^{N_h}(\boldsymbol{\mu})$ for some $\boldsymbol{\mu}$ in $\mathcal{D}$ – in terms of a relatively few retained snapshots. However, we must first ensure that we can choose a good combination of the available retained snapshots (Section 3.5.2), that we can represent the retained snapshots in a stable RB basis and efficiently obtain the associated RB basis coefficients (Section 3.5.3), and, finally, that we can choose our retained snapshots – in essence, the parameter sample $S_{N_{\max}}$ – optimally (Section 3.6).

3.5.2 *Galerkin projection*

For our particular class of equations, Galerkin projection is arguably the best approach. Given $\boldsymbol{\mu} \in \mathcal{D}$, evaluate

$$s_N^{N_h}(\boldsymbol{\mu}) = F(u_N^{N_h}(\boldsymbol{\mu}))$$

where $u_N^{N_h}(\boldsymbol{\mu}) \in V_N^{N_h} \subset V^{N_h}$, indicating here a general RB space (not necessarily a Lagrange space) with $V_N^{N_h}$, satisfies

$$a(u_N^{N_h}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = F(v) \quad \forall v \in V_N^{N_h} \tag{52}$$

We emphasize that our ultimate interest is the output prediction: the field variable serves as an intermediary. We immediately obtain (from Céa's lemma) the classical optimality

result in the energy norm (28)

$$\|u^{N_h}(\boldsymbol{\mu}) - u_N^{N_h}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \leq \inf_{w \in V_N^{N_h}} \|u^{N_h}(\boldsymbol{\mu}) - w\|_{\boldsymbol{\mu}} \tag{53}$$

in this norm, the Galerkin procedure automatically selects the *best* combination of snapshots. It is also readily derived that

$$s^{N_h}(\boldsymbol{\mu}) - s_N^{N_h}(\boldsymbol{\mu}) = \|u^{N_h}(\boldsymbol{\mu}) - u_N^{N_h}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}^2 \tag{54}$$

the output converges as the square of the energy error. Although this latter result depends critically on the compliance assumption, extension via adjoint approximations to the noncompliant case is possible; see Rozza *et al.* (2008).

We now consider the discrete equations associated with the Galerkin approximation (52). We must first choose an appropriate basis for our space: incorrect choice of the RB basis can lead to very poorly conditioned systems; this is immediately apparent in the Lagrange case – if $W_N^{N_h}$ provides rapid convergence, then, by construction, the snapshots of (51) will be increasingly co-linear as N increases. Toward this end, we apply the orthonormalization Gram–Schmidt process in the $(\cdot,\cdot)_V$ inner product to our snapshots $u^{N_h}(\boldsymbol{\mu}^n)$, $1 \leq n \leq N_{\max}$, to obtain mutually orthonormal functions $\zeta_n^{N_h}$, $1 \leq n \leq N_{\max}$: $(\zeta_n^{N_h}, \zeta_m^{N_h})_V = \delta_{nm}$, $1 \leq n, m \leq N_{\max}$, where δ_{nm} is the Kronecker-delta symbol. We then choose the sets $\{\zeta_n^{N_h}\}_{n=1,\ldots,N}$ as our bases for $W_N^{N_h}$, $1 \leq N \leq N_{\max}$.

We now insert

$$u_N^{N_h}(\boldsymbol{\mu}) = \sum_{m=1}^{N} u_{N\,m}(\boldsymbol{\mu}) \zeta_m^{N_h} \tag{55}$$

and $v = \zeta_n^{\mathcal{N}}$, $1 \leq n \leq N$, into (52) to obtain the RB algebraic system

$$\sum_{m=1}^{N} a(\zeta_m^{N_h}, \zeta_n^{N_h}; \boldsymbol{\mu})\, u_{N\,m}(\boldsymbol{\mu}) = F(\zeta_n^{N_h}), \quad 1 \leq n \leq N \tag{56}$$

for the RB coefficients $u_{N\,m}(\boldsymbol{\mu})$, $1 \leq m \leq N$; we can subsequently evaluate the RB output prediction as

$$s_N^{N_h}(\boldsymbol{\mu}) = \sum_{m=1}^{N} u_{N\,m}(\boldsymbol{\mu}) F(\zeta_m^{N_h}) \tag{57}$$

By using the Rayleigh quotient, it can be readily proven (Patera and Rozza 2007) that the condition number of the matrix $a(\zeta_m^{N_h}, \zeta_n^{N_h}; \boldsymbol{\mu})$, $1 \leq n, m \leq N$, is bounded by $\gamma^{\mathrm{e}}(\boldsymbol{\mu})/\alpha^{\mathrm{e}}(\boldsymbol{\mu})$, independently of N and N_h, owing to the orthogonality of the $\{\zeta_n^{N_h}\}$ and to (30) and (31).

3.5.3 Offline–online computational procedure

As anticipated for POD in Section 2.3, also the RB system (56) is nominally of small size: a set of N linear algebraic equations in N unknowns. However, the formation of the associated stiffness matrix, and indeed the right-hand side vector, involves entities $\zeta_n^{N_h}$, $1 \leq n \leq N$, associated with our N_h-dimensional FE approximation space. If we must invoke FE fields in order to form the RB stiffness matrix *for each new value of* $\boldsymbol{\mu}$, the marginal cost per input–output evaluation $\boldsymbol{\mu} \rightarrow s_N(\boldsymbol{\mu})$ will remain unacceptably large.

Fortunately, we can appeal to affine parameter dependence to construct very efficient offline–online procedures, as we now discuss. In particular, we note that system (56) can be expressed, thanks to (25), as

$$\sum_{m=1}^{N} \left(\sum_{q=1}^{Q} \Theta^q(\boldsymbol{\mu})\, a^q(\zeta_m^{N_h}, \zeta_n^{N_h}) \right) u_{N\,m}(\boldsymbol{\mu}) = F(\zeta_n^{N_h}), \quad 1 \leq n \leq N \tag{58}$$

We observe that ζ^{N_h} are now isolated in terms that are independent of $\boldsymbol{\mu}$ and hence that can be *precomputed* in an offline–online procedure.

In the offline stage, we first compute the $u^{N_h}(\boldsymbol{\mu}^n)$, $1 \leq n \leq N_{\max}$, and subsequently the $\zeta_n^{N_h}$, $1 \leq n \leq N_{\max}$; we then form and store the terms

$$F(\zeta_n^{N_h}), \quad 1 \leq n \leq N_{\max} \tag{59}$$

and

$$a^q(\zeta_m^{N_h}, \zeta_n^{N_h}), \quad 1 \leq n, m \leq N_{\max}, \quad 1 \leq q \leq Q \tag{60}$$

The offline operation count depends on $N_{\max}$, Q, and N_h.

In the online stage, we retrieve (60) to form

$$\sum_{q=1}^{Q} \Theta^q(\boldsymbol{\mu}) a^q(\zeta_m^{N_h}, \zeta_n^{N_h}), \quad 1 \leq n, m \leq N \tag{61}$$

We solve the resulting $N \times N$ stiffness system (58) to obtain $u_{N\,m}(\boldsymbol{\mu})$, $1 \leq m \leq N$; and finally we access (59) to evaluate the output (57). The online operation count is $O(QN^2)$ to perform the sum (61), $O(N^3)$ to invert (58) – note that the RB stiffness matrix is full – and finally $O(N)$ to effect the inner product (57). The online storage (the data archived in the offline stage) is due to the hierarchical condition (49) only $O(QN_{\max}^2) + O(N_{\max})$: for any given N, we may extract the necessary RB $N \times N$ matrices (respectively, N-vectors) as principal submatrices (respectively, principal subvectors) of the corresponding $N_{\max} \times N_{\max}$ (respectively, $N_{\max}$) quantities.

The online cost (operation count and storage) to evaluate $\boldsymbol{\mu} \rightarrow s_N^{N_h}(\boldsymbol{\mu})$ is thus independent of N_h. The implications are

twofold: first, *if* N is indeed small, we will achieve very fast response in the real-time and many-query contexts; second, we may choose N_h very conservatively – to make sure that the error between the exact and FE predictions is very small – without adversely affecting the online marginal cost.

We now turn to a more detailed discussion of sampling and (in Section 3.7) convergence in order to understand how, to a certain extent why, we can achieve high accuracy for N independent of N_h and indeed $N \ll N_h$.

3.6 Sampling strategies

We first indicate a few preliminaries, and then we provide one example of sampling strategies.

Let Ξ be a generic finite sample of points in $\mathcal{D}$ that will serve as surrogate for $\mathcal{D}$ in the calculation of errors (and, in Section 3.8, error bounds) over the parameter domain. Typically, these samples are chosen by Monte Carlo methods with respect to a uniform or log-uniform density: Ξ is, however, sufficiently large to ensure that the reported results are insensitive to further refinement of the parameter sample.

Given a function y: $\mathcal{D} \to \mathbb{R}$, we define

$$\|y\|_{L^\infty(\Xi)} = \max_{\boldsymbol{\mu}\in\Xi} |y(\boldsymbol{\mu})| \quad , \|y\|_{L^p(\Xi)} = \left(|\Xi|^{-1} \sum_{\boldsymbol{\mu}\in\Xi} |y|^p(\boldsymbol{\mu}) \right)^{1/p}$$

Here, $|\Xi|$ denotes the cardinality of the test sample Ξ. Given a function $z : \mathcal{D} \to V^{N_h}$ (or V^e), we then define

$$\|z\|_{L^\infty(\Xi;V)} = \max_{\boldsymbol{\mu}\in\Xi} \|z(\boldsymbol{\mu})\|_V ,$$

$$\|z\|_{L^p(\Xi;V)} - \left(|\Xi|^{-1} \sum_{\boldsymbol{\mu}\in\Xi} \|z(\boldsymbol{\mu})\|_V^p \right)^{1/p}$$

We denote the particular samples that shall serve to select our RB space – or train our RB approximation – by Ξ_{train}. The cardinality of Ξ_{train} will be denoted $|\Xi_{\text{train}}| = n_{\text{train}}$. We note that although the test samples Ξ serve primarily to understand and assess the quality of the RB approximation and *a posteriori* error estimators, the train samples Ξ_{train} serve to *generate* the RB approximation. The choice of n_{train} and Ξ_{train} thus have important offline and online computational implications.

We now illustrate a sample strategy particular to RB Lagrange spaces. The method can be viewed as a heuristic (more precisely, suboptimal) solution to the $L^\infty(\Xi_{\text{train}}; X)$ optimization problem analogous to the $L^2(\Xi_{\text{train}}; X)$ POD optimization problem (Section 2).

We are given Ξ_{train} and $N_{\max}$, as well as $S_1 = \{\boldsymbol{\mu}^1\}$, $W_1^{N_h\ \text{Greedy}} = \text{span}\{u^{N_h}(\boldsymbol{\mu}^1)\}$. In actual practice, we may set $N_{\max}$ either directly, or indirectly through a prescribed error tolerance. Then, for $N = 2, \ldots, N_{\max}$, we find

$$\boldsymbol{\mu}^N = \arg \max_{\boldsymbol{\mu}\in\Xi_{\text{train}}} \Delta_{N-1}(\boldsymbol{\mu})$$

set $S_N = S_{N-1} \cup \{\boldsymbol{\mu}^N\}$, and update $W_N^{N_h\ \text{Greedy}} = W_{N-1}^{N_h\ \text{Greedy}} \cup \text{span}\{u^{N_h}(\boldsymbol{\mu}^N)\}$. As we shall describe in detail in Section 3.8, $\Delta_N(\boldsymbol{\mu})$ is a sharp, asymptotically inexpensive *a posteriori* error bound for $\|u^{N_h}(\boldsymbol{\mu}) - u_N^{N_h}(\boldsymbol{\mu})\|_V$.

Roughly, at iteration N, this algorithm, called *greedy Lagrange RB algorithm*, appends to the *retained* snapshots that particular candidate snapshot – over all candidate snapshots $u^{N_h}(\boldsymbol{\mu})$, $\boldsymbol{\mu} \in \Xi_{\text{train}}$ – which is predicted by the *a posteriori* error bound to be the least well approximated by the RB prediction associated to $W_{N-1}^{N_h\ \text{Greedy}}$.

An analogous greedy procedure can be developed also in the energy norm (Rozza *et al.*, 2008) to build $W_N^{N_h\ \text{Greedy,en}}$, this norm being particularly relevant in the compliant case, since the error in the energy norm is directly related to the error in the output (Section 3.5.2).

3.7 Convergence of RB approximations

In this section we illustrate some convergence results for problems depending on one or several parameters.

3.7.1 A priori convergence theory: single parameter case

We present from Maday *et al.* (2002a,b), and Patera and Rozza (2007) an *a priori* theory for RB approximations associated with specific *nonhierarchical* Lagrange spaces $W^{N_h\ \ln}$, $1 \le N \le N_{\max}$, given by

$$W_N^{N_h\ \ln} = \text{span}\{u^{N_h}(\mu_N^n), 1 \le n \le N\} \tag{62}$$

for the parameter points given by

$$\mu_N^n = \mu^{\min} \exp\left\{ \frac{n-1}{N-1} \ln\left(\frac{\mu^{\max}}{\mu^{\min}} \right) \right\},$$
$$1 \le n \le N, \quad 1 \le N \le N_{\max} \tag{63}$$

We denote the corresponding RB approximation by $u_N^{N_h\ \ln}$.

The *a priori* theory described in the following suggests that the spaces (62) – which we shall denote "equi-ln" spaces – contain certainly optimality properties, though we shall observe that the more automatic greedy sample selection procedure does just as well (or perhaps even better for larger N). We note the analysis presented here in fact is relevant to a large class of single-parameter coercive problems.

If we consider a simple thermal block problem presented in Maday *et al.* (2002a) with the single parametric conductivity in only one of the two subblocks and the parameter domain $\mathcal{D} = [\mu^{\min}, \mu^{\max}] = [1/\sqrt{\mu_r}, \sqrt{\mu_r}]$, we have the following result:

Proposition 1. *Given general data F, we obtain that for any* $N \geq 1 + C_{\mu_r}$*, and for all* $\mu \in \mathcal{D}$*,*

$$\frac{\|u^{N_h}(\mu) - u_N^{N_h\ \ln}(\mu)\|_\mu}{\|u^{N_h}(\mu)\|_\mu} \leq \exp\left\{-\frac{N-1}{C_{\mu_r}}\right\} \tag{64}$$

where $C_{\mu_r} = [2e \ln \mu_r]$ *and* [ ] *returns the smallest integer greater than or equal to its real argument. Note we can directly derive from (64) and (54) a bound on the relative (compliant) output error.*

The proof is a "parameter" version of the standard (finite element) variational arguments, see Patera and Rozza (2007). In particular, we first invoke (53); we then take as our candidate w a high-order polynomial interpolant *in the parameter* μ of $u^{N_h}(\mu)$; we next apply the standard Lagrange interpolant remainder formula; finally, we appeal to an eigenfunction expansion to bound the parametric (sensitivity) derivatives and optimize the order of the polynomial interpolant. For the complete proof and more considerations, see Patera and Rozza (2007). We note that the RB convergence estimate (64), relative to the model problem we have considered, relies on parameter smoothness and not on the FE grid (through N_h); the exponent in the convergence rate depends on N and logarithmically on μ_r.

3.7.2 Multiparametric convergence

As already highlighted in the previous section, the key to RB convergence in higher parameter dimensions is the role of the PDE and field variable in determining appropriate sample points and combinations of snapshots. We illustrate the process schematically in Figure 2: the RB field approximation, via the PDE residual, yields the error bound; the error bound, in turn, facilitates the greedy selection of good sample points; the Galerkin projection then provides the optimal combination of retained snapshots; finally, the RB output approximation – application of the output functional – inherits the good properties of the RB field variable (54). The greedy sample points are quite nonintuitive and very far from the obvious (and inefficient) uniform distribution. In general, however, we do observe clustering near the boundaries of $\mathcal{D}$, as we might expect from classical approximation theory.

The computational success of the (implicit) complicated process described by Figure 2 is in fact also responsible for the failure, at present, to provide any general *a priori* convergence theory: we cannot construct an "optimal" approximant (like the piecewise polynomial interpolant in FE error analysis) since *a priori* we can neither predict an efficient sample nor construct an effective parametric interpolant.

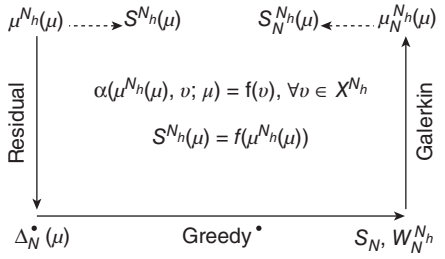

Figure 2. Schematic of the RB approximation process.

We can anticipate that, for a good set of points and from Galerkin a good combination of retained snapshots, we should obtain rapid convergence: $u^{N_h}(\boldsymbol{\mu}) \in V^{N_h}$ – the field we wish to approximate by the RB method – perforce resides on the parametrically induced low-dimensional *smooth* manifold $\mathcal{M}^{N_h} = \{u^{N_h}(\boldsymbol{\mu}) | \boldsymbol{\mu} \in \mathcal{D}\}$[3]; the essential role of parametric smoothness – already exploited in Section 3.7.1 for the single-parameter case – was identified in Fink and Rheinboldt (1983) and Porsching (1985). However, the fact that a good set of points (and hence a good Lagrange RB space) must exist, and that the greedy algorithm will identify any good set of points, is a topic of current research, see also Buffa *et al.* (2012), DeVore *et al.* (2013), Lassila *et al.* (2013), and Chen *et al.* (2014). Note in all cases we consider RB approximations associated with the spaces $W_N^{N_h} = W_N^{N_h\ \text{Greedy,en}}$.

A last remark about the spatial dimensionality, which plays little role in RB convergence: it follows that the relative efficiency of the RB approach – relative to direct FE evaluation – increases with increasing spatial dimension (see, e.g., Rozza *et al.*, 2008).

3.8 A posteriori error estimation

Effective *a posteriori* error bounds for the quantity of interest – our output – are crucial for both the efficiency and the reliability of RB approximations. As regards *efficiency* (related to the concept of "adaptivity" within the FE context), error bounds play a role in both the offline and online stages. In the greedy algorithm of Section 3.6, the application of error bounds (as surrogates for the actual error) permits significantly larger training samples $\Xi_{\text{train}} \subset \mathcal{D}$ at greatly reduced offline computational cost. These more extensive training samples in turn engender RB approximations that provide high accuracy at greatly reduced online

computational cost. The error bounds also serve directly in the online stage – to find the smallest RB dimension N that achieves the requisite accuracy – to further optimize online performance. In short, *a posteriori* error estimation allows control of the error, which in turn permits reduction of the computational effort. We should emphasize that *a posteriori* output error bounds are particularly important for RB approximations. First, RB approximations are ad hoc: each problem is different as regards discretization. Second, RB approximations are typically pre-asymptotic concerning the convergence error: we will choose N quite small – before any "tail" in the convergence rate. Third, the RB basis functions cannot be directly related to any spatial or temporal scales: physical intuition is of little value. And, fourth and finally, the RB approach is typically applied in the real-time context: there is no time for offline verification; errors are immediately manifested and often in deleterious ways. There is, thus, even greater need for *a posteriori* error estimation in the RB context than in the much more studied FE context.

The motivations for error estimation, in turn, place requirements on our error bounds. First, the error bounds must be *rigorous* – valid for all N and for all parameter values in the parameter domain $\mathcal{D}$: nonrigorous error "indicators" may suffice for adaptivity but not for reliability. Second, the bounds must be reasonably *sharp*: an overly conservative error bound can yield inefficient approximations (N too large) or suboptimal engineering results (unnecessary safety margins); design should be dictated by the output and not by the output error. And third, the bounds must be very *efficient*: the online operation count and storage to compute the RB error bounds – the marginal or asymptotic average cost – must be independent of N_h. Error bounds could also be developed for the gradient of the solution, for example, for velocity and pressure in potential flows (Rozza, 2011).

3.8.1 Preliminaries

The central equation in *a posteriori* theory is the error residual relationship (see Section 4.6.2). In particular, it follows from the problem statements for $u^{N_h}(\boldsymbol{\mu})$, (26), and $u_N^{N_h}(\boldsymbol{\mu})$, (52) that the error $e^{N_h}(\boldsymbol{\mu}) = e(\boldsymbol{\mu}) = u^{N_h}(\boldsymbol{\mu}) - u_N^{N_h}(\boldsymbol{\mu}) \in V_h^N$ satisfies

$$a(e(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = r(v; \boldsymbol{\mu}) \quad \forall\, v \in V^{N_h} \tag{65}$$

Here, $r(v; \boldsymbol{\mu}) \in (V^{N_h})'$, the dual space to V^{N_h}, is the residual

$$r(v; \boldsymbol{\mu}) = F(v; \boldsymbol{\mu}) - a(u_N^{N_h}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) \quad \forall\, v \in V^{N_h} \tag{66}$$

Equation (65) directly follows from the definition (66), $F(v; \boldsymbol{\mu}) = a(u^{N_h}(\boldsymbol{\mu}), v; \boldsymbol{\mu})$, $\forall\, v \in V^{N_h}$, bilinearity of a, and the definition of $e(\boldsymbol{\mu})$.

It will prove convenient to introduce the Riesz representation of $r(v; \boldsymbol{\mu})$ (see Theorem 2.1): $\hat{e}(\boldsymbol{\mu}) \in V^{N_h}$ satisfies

$$(\hat{e}(\boldsymbol{\mu}), v)_V = r(v; \boldsymbol{\mu}) \quad \forall\, v \in V^{N_h} \tag{67}$$

We can thus also write the error residual equation (65) as

$$a(e(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = (\hat{e}(\boldsymbol{\mu}), v)_V \quad \forall\, v \in V^{N_h} \tag{68}$$

It also follows that

$$\|r(\,\cdot\,; \boldsymbol{\mu})\|_{(V^{N_h})'} = \sup_{v \in V^{N_h}} \frac{r(v; \boldsymbol{\mu})}{\|v\|_V} = \|\hat{e}(\boldsymbol{\mu})\|_V \tag{69}$$

The evaluation of the dual norm of the residual through the Riesz representation theorem is central to the offline–online procedures developed in Section 3.8.3.

We recall the definition of the exact and FE coercivity constants, (30) and (31), respectively. We shall require a lower bound to the coercivity constant $\alpha^{N_h}(\boldsymbol{\mu})$, $\alpha_{\mathrm{LB}}^{N_h}\colon \mathcal{D} \to \mathbb{R}$, such that (i) $0 < \alpha_{\mathrm{LB}}^{N_h}(\boldsymbol{\mu}) \le \alpha^{N_h}(\boldsymbol{\mu})$ $\forall\, \boldsymbol{\mu} \in \mathcal{D}$, and (ii) the online computational time to evaluate $\boldsymbol{\mu} \to \alpha_{\mathrm{LB}}^{N_h}(\boldsymbol{\mu})$ is independent of N_h. In Section 3.8.4 we summarize a basic methodology (Huynh *et al.*, 2007) to construct the requisite lower bound. The general case is addressed in Hesthaven *et al.* (2015).

3.8.2 Error bounds

We define error estimators for the energy norm and output as

$$\Delta_N^{\mathrm{en}}(\boldsymbol{\mu}) = \|\hat{e}(\boldsymbol{\mu})\|_V / (\alpha_{\mathrm{LB}}^{N_h}(\boldsymbol{\mu}))^{1/2}$$

and

$$\Delta_N^{s}(\boldsymbol{\mu}) = \|\hat{e}(\boldsymbol{\mu})\|_V^2 / \alpha_{\mathrm{LB}}^{N_h}(\boldsymbol{\mu})$$

respectively. We next introduce the effectivities associated with these error estimators as

$$\eta_N^{\mathrm{en}}(\boldsymbol{\mu}) = \Delta_N^{\mathrm{en}}(\boldsymbol{\mu}) / |\!|\!| u^{N_h}(\boldsymbol{\mu}) - u_N^{N_h}(\boldsymbol{\mu}) |\!|\!|_{\boldsymbol{\mu}}$$

and

$$\eta_N^{s}(\boldsymbol{\mu}) = \Delta_N^{s}(\boldsymbol{\mu}) / (s^{N_h}(\boldsymbol{\mu}) - s_N^{N_h}(\boldsymbol{\mu}))$$

respectively.

Clearly, the effectivities are a measure of the quality of the proposed estimator: for rigor, we shall insist upon effectivities ≥ 1; for sharpness, we desire effectivities as close to unity as possible. It has been reported, for example, in

Rozza *et al.* (2008) and Patera and Rozza (2007), that for any $N = 1, \ldots, N_{\max}$, the effectivities satisfy

$$1 \le \eta_N^{\text{en}}(\boldsymbol{\mu}) \le \sqrt{\frac{\gamma^{\text{e}}(\boldsymbol{\mu})}{\alpha_{\text{LB}}^{N_h}(\boldsymbol{\mu})}} \quad \forall\, \boldsymbol{\mu} \in \mathcal{D} \tag{70}$$

$$1 \le \eta_N^{s}(\boldsymbol{\mu}) \le \frac{\gamma^{\text{e}}(\boldsymbol{\mu})}{\alpha_{\text{LB}}^{N_h}(\boldsymbol{\mu})} \quad \forall\, \boldsymbol{\mu} \in \mathcal{D} \tag{71}$$

Similar results can be obtained for $\Delta_N(\mu)$, the *a posteriori* error bound in the V norm.

It is important to observe that the effectivity upper bounds, (70) and (71), are *independent* of N, and hence stable with respect to *RB refinement*. Furthermore, it is sometimes possible (Section 3.8.4) to provide a rigorous lower bound for $\alpha_{\text{LB}}^{N_h}(\boldsymbol{\mu})$ that depends only on $\boldsymbol{\mu}$: in this case we obtain an upper bound for the effectivity, which is not only independent of N but also independent of N_h, and hence stable with respect to *FE refinement*.

3.8.3 Offline–online computational procedure

The error bounds of the previous section are of no utility without an accompanying offline–online computational approach.

The computationally crucial component of all the error bounds of the previous section is $\|\hat{e}(\boldsymbol{\mu})\|_V$, the dual norm of the residual.

To develop an offline–online procedure for the dual norm of the residual, we first expand the residual (66) according to (55) and (25):

$$\begin{aligned} r(v;\boldsymbol{\mu}) &= F(v) - a(u_N^{N_h}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) \\ &= F(v) - a\left(\sum_{n=1}^{N} u_{Nn}(\boldsymbol{\mu})\, \zeta_n^{N_h}, v; \boldsymbol{\mu}\right) \\ &= F(v) - \sum_{n=1}^{N} u_{Nn}(\boldsymbol{\mu})\, a(\zeta_n^{N_h}, v; \boldsymbol{\mu}) \\ &= F(v) - \sum_{n=1}^{N} u_{Nn}(\boldsymbol{\mu}) \sum_{q=1}^{Q} \Theta^q(\boldsymbol{\mu})\, a^q(\zeta_n^{N_h}, v) \end{aligned} \tag{72}$$

If we insert (72) in (67) and apply linear superposition, we obtain

$$(\hat{e}(\boldsymbol{\mu}), v)_V = F(v) - \sum_{q=1}^{Q} \sum_{n=1}^{N} \Theta^q(\boldsymbol{\mu})\, u_{Nn}(\boldsymbol{\mu})\, a^q(\zeta_n^{N_h}, v)$$

or

$$\hat{e}(\boldsymbol{\mu}) = C + \sum_{q=1}^{Q} \sum_{n=1}^{N} \Theta^q(\boldsymbol{\mu})\, u_{Nn}(\boldsymbol{\mu})\, \mathcal{L}_n^q$$

where $C \in V^{N_h}$, $(C, v)_V = f(v)\ \forall\, v \in V^{N_h}$, and $\mathcal{L}_n^q, \in V^{N_h}$ satisfies $(\mathcal{L}_n^q, v)_V = -a^q(\zeta_n^{N_h}, v) \quad \forall\, v \in V^{N_h}, \quad 1 \le n \le N$, $1 \le q \le Q$.

We thus obtain

$$\begin{aligned} &\|\hat{e}(\boldsymbol{\mu})\|_V^2 \\ &= (C, C)_V + \sum_{q=1}^{Q} \sum_{n=1}^{N} \Theta^q(\boldsymbol{\mu})\, u_{Nn}(\boldsymbol{\mu}) \\ &\left\{ 2(C, \mathcal{L}_n^q)_V + \sum_{q'=1}^{Q} \sum_{n'=1}^{N} \Theta^{q'}(\boldsymbol{\mu})\, u_{Nn'}(\boldsymbol{\mu})\, (\mathcal{L}_n^q, \mathcal{L}_{n'}^{q'})_V \right\} \end{aligned} \tag{73}$$

from which we can directly calculate the requisite dual norm of the residual through (69).

The offline–online decomposition is now clear. In the offline stage we form the parameter-independent quantities. In particular, we compute the FE solutions $C, \mathcal{L}_n^q$, $1 \le n \le N_{\max}$, $1 \le q \le Q$, and form/store $(C, C)_V$, $(C, \mathcal{L}_n^q)_V$, $(\mathcal{L}_n^q, \mathcal{L}_{n'}^{q'})_V$, $1 \le n, n' \le N_{\max}$, $1 \le q, q' \le Q$. Note that a single matrix factorization suffices to obtain all $1 + QN_{\max}$ FE solutions. The offline operation count depends on $N_{\max}$, Q, and N_h.

In the online stage, given any "new" value of $\boldsymbol{\mu}$ and therefore $\Theta^q(\boldsymbol{\mu})$, $1 \le q \le Q$, $u_{N\,n}(\boldsymbol{\mu})$, $1 \le n \le N$, we simply retrieve the stored quantities $(C, C)_V$, $(C, \mathcal{L}_n^q)_V$, $(\mathcal{L}_n^q, \mathcal{L}_{n'}^{q'})_V$, $1 \le n, n' \le N$, $1 \le q, q' \le Q$, and then evaluate the sum (73). The online operation count, and hence also the marginal cost and asymptotic average cost, is $O(Q^2N^2)$ and is *independent of* N_h.[4] Note that with hierarchical spaces the necessary quantities for any $N \in \{1, \ldots, N_{\max}\}$ can be simply extracted from the corresponding quantities for $N = N_{\max}$.

3.8.4 Lower bounds for the coercivity constant

As introduced in Section 3.8, our *a posteriori* error analysis of reduced basis approximations to affinely parametrized partial differential equations requires a lower bound for the coercivity constant or a more general stability factor (inf–sup) in noncoercive problems (Hesthaven *et al.*, 2015)

In essence, the discrete parametrized coercivity constant (31) is a generalized minimum eigenvalue (Patera and Rozza, 2007). There are many classical techniques for the estimation of minimum eigenvalues or minimum singular values.

An approach to the construction of lower bounds for coercivity (and, in the noncoercive case, inf–sup stability) constants is the successive constraint method (SCM) introduced in Huynh *et al.* (2007). The method – based on an offline–online strategy relevant in the many-query and real-time context – reduces the online (real-time) calculation to a small linear program for which the operation count is *independent* of N_h, see, for basic details, Rozza *et al.* (2008). Quarteroni *et al.* (2011) and Hesthaven *et al.* (2015) provide

a generalization to other classes of problems, the latter containing also a comparison with simpler methodologies (like the θ-method, see Rozza, 2005).

3.9 Historical background, state of the art, extensions, and perspectives

We provide some background and perspectives for RB methods. As we have seen, RB discretization is, in brief, a Galerkin projection on an N-dimensional approximation space. Initial work grew out of two related streams of inquiry: from the need for more effective, many-query design evaluation (Fox and Miura, 1971); and from the need for more efficient parameter continuation methods for nonlinear problems depending on a parameter (Almroth *et al.*, 1978; Noor *et al.*, 1980, 1984; Noor, 1981, 1982).

These early approaches were soon extended to (i) general finite-dimensional systems as well as certain classes of PDEs (and ODEs), see Barrett and Reddien (1995), Fink and Rheinboldt (1983), Lee (1991), Noor *et al.* (1984), Porsching and Lee (1987), Rheinboldt (1981), Rheinboldt (1993), and (ii) a variety of different reduced basis approximation spaces – in particular Taylor and Lagrange (Porsching, 1985) and more recently Hermite expansions (Ito and Ravindran, 1998). Further extensions were concerned with different applications and classes of equations, especially fluid dynamics with the incompressible Navier–Stokes equations, see, for example, Peterson (1989).

In these early methods, the approximation spaces were rather local and typically low-dimensional in parameter (often a single parameter). This was primarily due to the absence of *a posteriori* error estimators and effective sampling procedures. Indeed, in more global higher dimensional parameter domains, the ad hoc reduced basis predictions "far" from any sample points cannot necessarily be trusted, and hence *a posteriori* error estimators are crucial to reliability. Moreover, sophisticated sampling strategies for parameters are crucial to convergence and computational efficiency.

Much current effort is thus devoted to the development of (i) *a posteriori* error estimation procedures and in particular rigorous error bounds for outputs of interest, and (ii) effective sampling strategies, in particular for the many-parameter case. The *a posteriori* error bounds are of course indispensable for rigorous certification of any particular reduced basis output prediction. However, the error estimators can also play an important role in efficient and effective sampling procedures: the inexpensive error bounds allow one, first, to explore much larger subsets of the parameter domain in search of most representative or best "snapshots"; and, second, to determine when we have *just enough* basis functions.

We note here that greedy sampling methods are similar in objective to more well-known POD methods of Section 2. Both have been widely applied in the (multidimensional) parameter domain. POD techniques can even be combined within the parametric RB context, see Haasdonk and Ohlberger (2008), Nguyen *et al.* (2009). A brief comparison of greedy and POD approaches – computational cost and performance – is reported in Patera and Rozza (2007) and Rozza *et al.* (2008).

The reduced basis approach can also be readily applied to the more general case of affine linear elliptic *noncoercive* problems (see, e.g., Quarteroni *et al.*, 2011; Hesthaven *et al.*, 2015). The special issues associated with *saddle-point problems*, in particular the Stokes equations of incompressible flow, have been addressed in Rozza and Veroy (2007), Gerner and Veroy (2012), and Rozza *et al.* (2013).

The exploration of the "parameter + time" framework in the context of affine linear parabolic PDEs – such as the heat equation and the convection-diffusion equation – is explained in Grepl and Patera (2005), the review by Quarteroni *et al.* (2011), and in the book by Hesthaven *et al.* (2015).

The reduced basis methodology, in both the elliptic and parabolic cases, can also be extended to problems with nonaffine parametric variation. The strategy consists of reducing the nonaffine operator and data to an approximate affine form, and then apply the methods developed for affine operators described in this section. However, this reduction must be done efficiently in order to avoid a proliferation of parametric functions and a corresponding degradation of online response time. This extension is based on the already mentioned *empirical interpolation method* Barrault *et al.* (2004): a collateral RB space for the nonaffine coefficient functions; an interpolation system that avoids costly (N_h-dependent) projections; and several *a posteriori* error estimators. A review of the empirical interpolation method within the context of RB treatment of elliptic and parabolic PDEs with nonaffine coefficient functions is considered in the book by Hesthaven *et al.* (2015); the resulting approximations preserve the usual offline–online efficiency – the complexity of the online stage is independent of N_h.

The reduced basis approach and associated offline–online procedures can be applied without serious computational difficulties to quadratic nonlinearities (Veroy *et al.*, 2003; Canuto *et al.*, 2009), for example, but not limited, to the stationary incompressible Navier–Stokes equations: suitable stable approximations are considered in Nguyen *et al.* (2005), Veroy and Patera (2005), Quarteroni and Rozza (2007), and Deparis and Rozza (2009). For a more complete overview of current developments, see the focus introduction (Chapter 1) in Hesthaven *et al.* (2015), and references therein, especially for current approaches to multiscale and

multiphysics problems, applications to domain decomposition, data assimilation, optimal control, inverse problems, and uncertainty quantification, as well as more complex (geometrical) parametrizations. Needless to say, these are fast growing research areas in which substantial progress is already being made for the coming years.

4 PROPER GENERALIZED DECOMPOSITION

4.1 Motivation

Most of the existing model reduction techniques proceed by extracting a suitable reduced basis and then projecting on it the problem solution. Thus, the reduced basis construction precedes its use in the solution procedure, and one must be careful on the suitability of a particular reduced basis when employed for representing the solution of a particular problem. This issue disappears if the approximation basis is constructed at the same time as when the problem is solved. Thus, each problem has its associated basis in which its solution is expressed. One could consider few terms in its approximation, leading to a reduced representation, or all the terms needed for approximating the solution up to a certain accuracy level. The PGD, described in general terms in the next section, proceeds in this manner.

When calculating the transient solution of a generic problem $u(x,t)$, we usually consider a given basis of space functions $N_i(x)$, $i = 1, \dots, N$, the so-called shape functions within the finite element framework, and approximate the problem solution as

$$u(x,t) \approx \sum_{i=1}^{N} a_i(t) N_i(x) \tag{74}$$

which implies a space–time separated representation where the time-dependent coefficients $a_i(t)$ are unknown at each time (when proceeding incrementally) and the space functions $N_i(x)$ are given *a priori*, for example, polynomial basis. POD and reduced bases methodologies consider a reduced basis $\phi_i(x)$ for approximating the solution instead of using the generic functions $N_i(x)$. The former are expected to be more suitable for approximating the problem at hand. Thus, it results in

$$u(x,t) \approx \sum_{i=1}^{R} b_i(t) \phi_i(x) \tag{75}$$

where, in general, $R \ll N$. Again, (75) represents a space–time separated representation where the time-dependent coefficient must be calculated at each time during the incremental solution procedure.

Inspired from these results one could consider the general space-time separated representation

$$u(x,t) \approx \sum_{i=1}^{N} X_i(x) \cdot T_i(t) \tag{76}$$

where now neither the time-dependent functions $T_i(t)$ nor the space functions $X_i(x)$ are *a priori* known. Both will be computed on the flight when solving the problem.

As soon as one postulates that the solution of a transient problem can be expressed in the separated form (76) whose approximation functions $X_i(x)$ and $T_i(t)$ will be determined during the problem solution, one could take a step forward and assume that the solution of a multidimensional problem $u(x_1, \dots, x_d)$ could be found in the separated form

$$u(x_1, x_2, \dots, x_d) \approx \sum_{i=1}^{N} X_i^1(x_1) \cdot X_i^2(x_2) \cdots X_i^d(x_d) \tag{77}$$

and, even more, expressing the 3D solution $u(x,y,z)$ as a finite sum decomposition involving lower dimensional functions

$$u(x,y,z) \approx \sum_{i=1}^{N} X_i(x) \cdot Y_i(y) \cdot Z_i(z) \tag{78}$$

or

$$u(x,y,z) \approx \sum_{i=1}^{N} X_i(x,y) \cdot Z_i(z) \tag{79}$$

and the solution of a parametric problem $u(\mathbf{x}, t, p_1, \dots, p_{\wp})$ as

$$u(\mathbf{x}, t, p_1, \dots, p_{\wp}) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot T_i(t) \cdot \prod_{k=1}^{\wp} P_i^k(p_k) \tag{80}$$

The performances of all these separated representations are quite impressive in many cases; however, the key point when considering such separated representations lies in the algorithm to be used for calculating the different functions that they involve: $T_i(t)$, $X_i(x)$, $P_i(p)$. Both questions will be addressed in this section.

4.2 Illustrating the simplest separated representation constructor

In order to illustrate the simplest procedure for constructing the separated representation, we consider the one-dimensional heat transfer equation involving the temperature field $u(x,t)$

$$\frac{\partial u}{\partial t} - k \frac{\partial^2 u}{\partial x^2} = f \tag{81}$$

defined in the space–time domain $\Omega = \Omega_x \times \Omega_t = (0, L) \times (0, \tau]$. The diffusivity k and source term f are assumed constant. We specify homogeneous initial and boundary conditions, that is, $u(x, t = 0) = u(x = 0, t) = u(x = L, t) = 0$. More details and more complex scenarios can be found in Chinesta *et al.* (2013a).

The weighted residual form of (81) reads

$$\int_{\Omega_x \times \Omega_t} u^* \left(\frac{\partial u}{\partial t} - k \frac{\partial^2 u}{\partial x^2} - f \right) \, \mathrm{d}x \, \mathrm{d}t = 0 \tag{82}$$

for all suitable test functions u^*.

Our objective is to obtain a PGD approximate solution in the separated form

$$u(x, t) \approx \sum_{i=1}^{N} X_i(x) \cdot T_i(t) \tag{83}$$

We do so by computing each term of the expansion at each step of an enrichment process, until a suitable stopping criterion is met.

4.2.1 Progressive construction of the separated representation

At enrichment step n, the $n-1$ first terms of the PGD approximation (83) are known:

$$u^{n-1}(x, t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) \tag{84}$$

We now wish to compute the next term $X_n(x) \cdot T_n(t)$ to get the enriched PGD solution

$$\begin{aligned} u^n(x, t) &= u^{n-1}(x, t) + X_n(x) \cdot T_n(t) \\ &= \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n(x) \cdot T_n(t) \end{aligned} \tag{85}$$

One must thus solve a nonlinear problem for the unknown functions $X_n(x)$ and $T_n(t)$ by means of a suitable iterative scheme. The simplest strategy consists of an alternated direction fixed point algorithm.

At enrichment step n, the PGD approximation $u^{n,p}$ obtained at iteration p is given by

$$u^{n,p}(x, t) = u^{n-1}(x, t) + X_n^p(x) \cdot T_n^p(t) \tag{86}$$

Starting from an arbitrary initial guess $T_n^0(t)$, the alternating direction strategy computes $X_n^p(x)$ from $T_n^{p-1}(t)$, and then $T_n^p(t)$ from $X_n^p(x)$. These nonlinear iterations proceed until reaching a fixed point within a user-specified tolerance ϵ, that is

$$\| X_n^p(x) \cdot Y_n^p(y) - X_n^{p-1}(x) \cdot Y_n^{p-1}(y) \| < \epsilon \tag{87}$$

where $\| \cdot \|$ is a suitable norm. The enrichment step n thus ends with the assignments $X_n(x) \leftarrow X_n^p(x)$ and $T_n(t) \leftarrow T_n^p(t)$.

The enrichment process itself stops when an appropriate measure of error $\mathcal{E}(n)$ becomes small enough, that is, $\mathcal{E}(n) < \tilde{\epsilon}$. One can apply different stopping criteria discussed in Ammar *et al.* (2010a), Ladeveze and Chamoin (2011), Moitinho de Almeida (2013), Chinesta *et al.* (2013a), and Nadal *et al.* (2015).

Let us look at one particular alternating direction iteration at a given enrichment step.

4.2.2 Alternating direction strategy

Each iteration of the alternating direction scheme consists of the following two steps:

- *Calculating $X_n^p(x)$ from $T_n^{p-1}(t)$.*
  At this stage, the approximation is given by

$$u^n(x, t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n^p(x) \cdot T_n^{p-1}(t) \tag{88}$$

  where all functions but $X_n^p(x)$ are known. The simplest choice for the weight function u^* in (82) is

$$u^*(x, t) = X_n^*(x) \cdot T_n^{p-1}(t) \tag{89}$$

  which amounts to considering a Galerkin formulation of the diffusion problem. Introducing (88) and (89) into (82), we obtain

$$\begin{aligned} &\int_{\Omega_x \times \Omega_t} X_n^* T_n^{p-1} \left(X_n^p \frac{\mathrm{d}T_n^{p-1}}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} T_n^{p-1} \right) \, \mathrm{d}x \, \mathrm{d}t \\ &\quad = - \int_{\Omega_x \times \Omega_t} X_n^* T_n^{p-1} \sum_{i=1}^{n-1} \left(X_i \frac{\mathrm{d}T_i}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} T_i \right) \, \mathrm{d}x \, \mathrm{d}t \\ &\qquad + \int_{\Omega_x \times \Omega_t} X_n^* T_n^{p-1} \, f \, \mathrm{d}x \, \mathrm{d}t \end{aligned} \tag{90}$$

  As all functions of time t are known, we can evaluate the following integrals:

$$\begin{cases} \alpha^x = \int_{\Omega_t} (T_n^{p-1}(t))^2 \, \mathrm{d}t \\ \beta^x = \int_{\Omega_t} T_n^{p-1}(t) \dfrac{\mathrm{d}T_n^{p-1}(t)}{\mathrm{d}t} \, \mathrm{d}t \\ \gamma_i^x = \int_{\Omega_t} T_n^{p-1}(t) T_i(t) \, \mathrm{d}t \\ \delta_i^x = \int_{\Omega_t} T_n^{p-1}(t) \dfrac{\mathrm{d}T_i(t)}{\mathrm{d}t} \, \mathrm{d}t \\ \xi^x = \int_{\Omega_t} T_n^{p-1}(t) \; f \, \mathrm{d}t \end{cases} \tag{91}$$

Equation (90) then takes the form

$$\int_{\Omega_x} X_n^* \left(-k\alpha^x \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} + \beta^x X_n^p \right) \mathrm{d}x = \int_{\Omega_x} X_n^* \sum_{i=1}^{n-1} \left(k\gamma_i^x \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} - \delta_i^x X_i \right) \mathrm{d}x + \int_{\Omega_x} X_n^* \xi^x \, \mathrm{d}x \tag{92}$$

This defines a one-dimensional boundary value problem (BVP), which is readily solved by means of a standard finite element method to obtain an approximation of the function X_n^p. As another option, one can go back to the associated strong form

$$-k\alpha^x \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} + \beta^x X_n^p = \sum_{i=1}^{n-1} \left(k\gamma_i^x \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} - \delta_i^x X_i \right) + \xi^x \tag{93}$$

and then solve it using any suitable numerical method, such as finite differences, for example. The strong form (93) is a second-order differential equation for X_n^p because of the fact that the original diffusion equation (81) involves a second-order x-derivative of the unknown field u. The homogeneous Dirichlet boundary conditions $X_n^p(x = 0) = X_n^p(x = L) = 0$ are readily specified with either weak or strong formulations.

- *Calculating $T_n^p(t)$ from the just-computed $X_n^p(x)$.*
The procedure mirrors what we have just done. It suffices to exchange the roles played by the relevant functions of x and t. The current PGD approximation reads

$$u^n(x,t) = \sum_{i=1}^{n-1} X_i(x) \cdot T_i(t) + X_n^p(x) \cdot T_n^p(t) \tag{94}$$

where all functions are known except $T_n^p(t)$. With the weighting function

$$u^*(x,t) = X_n^p(x) \cdot T_n^*(t) \tag{95}$$

the weighted residual form (82) becomes

$$\begin{aligned} \int_{\Omega_x \times \Omega_t} X_n^p T_n^* \left(X_n^p \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_n^p}{\mathrm{d}x^2} T_n^p \right) \mathrm{d}x \, \mathrm{d}t \\ = - \int_{\Omega_x \times \Omega_t} X_n^p T_n^* \sum_{i=1}^{n-1} \left(X_i \frac{\mathrm{d}T_i}{\mathrm{d}t} - k \frac{\mathrm{d}^2 X_i}{\mathrm{d}x^2} T_i \right) \mathrm{d}x \, \mathrm{d}t \\ + \int_{\Omega_x \times \Omega_t} X_n^p T_n^* \; f \, \mathrm{d}x \, \mathrm{d}t \end{aligned} \tag{96}$$

Since all functions of x are known, we can perform the following integrals:

$$\begin{cases} \alpha^t = \int_{\Omega_x} (X_n^p(x))^2 \, \mathrm{d}x \\ \beta^t = \int_{\Omega_x} X_n^p(x) \dfrac{\mathrm{d}^2 X_n^p(x)}{\mathrm{d}x^2} \, \mathrm{d}x \\ \gamma_i^t = \int_{\Omega_x} X_n^p(x) X_i(x) \, \mathrm{d}x \\ \delta_i^t = \int_{\Omega_x} X_n^p(x) \dfrac{\mathrm{d}^2 X_i(x)}{\mathrm{d}x^2} \, \mathrm{d}x \\ \xi^t = \int_{\Omega_x} X_n^p(x) \; f \, \mathrm{d}x \end{cases} \tag{97}$$

Equation (96) then becomes

$$\int_{\Omega_t} T_n^* \left(\alpha^t \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k\beta^t T_n^p \right) \mathrm{d}t = \int_{\Omega_t} T_n^* \sum_{i=1}^{n-1} \left(-\gamma_i^t \frac{\mathrm{d}T_i}{\mathrm{d}t} + k\delta_i^t T_i \right) \mathrm{d}t + \int_{\Omega_t} T_n^* \xi^t \, \mathrm{d}t \tag{98}$$

We have thus obtained an initial value problem (IVP) for the function T_n^p. The weighted residual form (98) can be solved by means of any stabilized finite element scheme (e.g., discontinuous Galerkin). The associated strong form reads

$$\alpha^t \frac{\mathrm{d}T_n^p}{\mathrm{d}t} - k\beta^t T_n^p = \sum_{i=1}^{n-1} \left(-\gamma_i^t \frac{\mathrm{d}T_i}{\mathrm{d}t} + k\delta_i^t T_i \right) + \xi^t \tag{99}$$

Since the original diffusion equation involves a first-order derivative of u with respect to t, we have thus obtained a first-order ODE for T_n^p. Any classical numerical technique can be used to solve it. The initial condition $T_n^p(t = 0) = 0$ is readily specified with either weak or strong form.

4.2.3 The numerical analysis viewpoint

There exist several approaches to the numerical analysis of PGD methods. They combine the existence of a best approximation and a greedy algorithm (Ammar *et al.*, 2010b). A first theoretical study by Le Bris *et al.* (2009) addressed

symmetric elliptic problems and its connection to greedy algorithms from nonlinear approximation theory deeply analyzed by DeVore and Temlyakov (1996). In Figueroa and Suli (2011), the authors extend the convergence analysis of the pure greedy and orthogonal greedy algorithms considered in Le Bris *et al.* (2009) to the technically more complicated case where the Laplace operator is replaced by a high-dimensional Ornstein–Uhlenbeck operator with unbounded drift.

General elliptic problems were considered in Falco and Nouy (2011a). The extension of these results to nonlinear symmetric coercive problems was considered by Cancès *et al.* (2011), which also contains specific results when the algorithm is applied to finite-dimensional problems. Current research tracks include, for example, the extension of the convergence analysis to nonsymmetric problems and the analysis of the rate of convergence for nonlinear equations. Falco and Nouy (2011b) provide a mathematical analysis of a family of progressive and updated PGDs for a particular class of convex optimization problems in reflexive tensor Banach spaces. These results lead to a generalization of the concept of SVD; concretely, a constrained version of the SVD is proposed in Nouy and Falco (2011).

Despite these advances accomplished in the mathematical foundations of PGD-based techniques, most questions concerning the optimality of the constructed separated representations, the convergence of the constructor for nonsymmetric and/or nonlinear operators, and error bounds, among many other issues, remain open (Cancès *et al.*, 2013; Falco *et al.*, 2013).

Other important source of theoretical questions is the relationship between the time-dependent PGD framework and the time-dependent Dirac–Frenkel variational method first used by Dirac in 1930. This method plays a similar fundamental role for the time-dependent Schrodinger equation as the Rayleigh–Ritz variational principle for the Schrodinger eigenvalue problem (see Lubich, 2008; Ehrlacher, 2014).

4.3 Nonincremental versus incremental time integrations

It is useful to reflect on the considerable difference between the above PGD strategy and traditional incremental time integration schemes. Indeed, the PGD allows for a *nonincremental* solution of time-dependent problems. Let Q_n denote the number of nonlinear iterations of the alternating direction algorithm required to compute the new term $X_n(x) \cdot T_n(t)$ at enrichment step n. Then, the entire PGD procedure to obtain the N-term approximation (83) involves the solution of a total of $Q = (Q_1 + \cdots + Q_N)$ *decoupled* one-dimensional BVPs and IVPs. The BVPs are defined over the space domain Ω_x, and their computational complexity scales with the size of the one-dimensional mesh used to discretize them. The IVPs are defined over the time interval Ω_t, and their complexity is usually negligible compared to that of the BVPs, even when extremely small time steps are used for their discretization.

This is vastly different from a standard incremental solution procedure. If P is the total number of time steps for the complete simulation, that is, $P = \tau/\Delta t$, an incremental procedure involves the solution of a BVP in Ω_x at each time step, that is, a total of P BVPs. This can be a very large number indeed, as the time step Δt must be chosen small enough to guarantee the stability of the numerical scheme. In the case of 3D transient problems, the CPU time savings when using PGD could be in some cases many orders of magnitude.

Space–time separated representations were considered in the works of Ladeveze from the 1980s and constitute one of the key bricks of the nonlinear and nonincremental solver called LArge Time INcrement method – LATIN – (Ladeveze, 1985, 1989, 1999). During the past three decades, many works were successfully accomplished by the Cachan group around Ladeveze to improve the space–time separated representation and its applicability in complex nonlinear problems involving large displacements (Ladeveze, 1996; Boucard *et al.*, 1997), large size multiscale models (Ladeveze *et al.*, 2010; Neron and Ladeveze, 2010), homogenization (Ladeveze and Nouy, 2003; Nouy and Ladeveze, 2004; Cremonesi *et al.*, 2013), domain decomposition (Blanze *et al.*, 1996; Champaney *et al.*, 1997; Ladeveze and Dureisseix, 1998; Ladeveze *et al.*, 2007), problems involving parameter variability (Boucard and Ladeveze, 1999), multiphysics (Dureisseix *et al.*, 2003a,b; Neron and Dureisseix, 2008a,b), among many others topics. Recently, Heyberger *et al.* (2013) and Neron *et al.* (2015) adapted the standard LATIN formalism for addressing parametric models.

Dynamics in the medium frequency range was another topic widely considered by the Cachan group for long time, where space–time representations were replaced by space–frequency descriptions (not fully separated) within the so-called TVCR techniques (Riou *et al.*, 2013; Barbarulo *et al.*, 2014a,b).

Other than the prolific contribution of the Cachan group to space–time (and its space–frequency counterpart) separated representations, these representations were also considered in other works. Thus, Boucinha *et al.* (2013, 2014) proposed the use space–time representations for solving dynamical problems formulated in a displacement–velocity mixed framework and where advanced separated representation constructors were considered. In Ammar *et al.* (2015), the authors propose a space–time representation within the PGD framework applied to the integral formulation of viscoelastic behaviors. The issue related to multiple

scales in time was addressed in Ammar *et al.* (2011a) and Chinesta *et al.* (2010a), where multi-time-separated representations and advanced coupling strategies were, respectively, proposed. Other works focusing on time multiscale were considered in Hammoud *et al.* (2014). Space–time separated representations were considered in the solution of fluid flow problems in Aghighi *et al.* (2013) and Leblond and Allery (2014). A real-time solid dynamics approach was proposed in Gonzalez *et al.* (2014) based on the use of a parametric solution involving initial conditions. Different constructors of the space–time separated representation were analyzed by Nouy (2010a). In Bonithon *et al.* (2011), the PGD was introduced within the boundary element method framework avoiding the use of space–time kernels, and then enhancing the discretization efficiency. The LATIN nonlinear solver combined with separated representations was also considered in Giacoma *et al.* (2015) for treating contact problems. Electromagnetic models were addressed in Henneron and Clenet (2013, 2015).

On the other hand, fully separated space–frequency representation with a parametric approach was employed in the linear and nonlinear case in Modesto *et al.* (2015) and Germoso *et al.* (2016). In Aguado *et al.* (2015), space–frequency representations were extended for treating parametric parabolic problems and were combined with the reciprocity principle for attaining real-time performances in the case of moving thermal loads.

4.4 Space separation

Sometimes the spatial domain Ω, assumed three dimensional, can be fully or partially separated, and consequently it can be expressed as $\Omega = \Omega_x \times \Omega_y \times \Omega_z$ or $\Omega = \Omega_{xy} \times \Omega_z$, respectively. The first decomposition is related to hexahedral domains, whereas the second one is related to plate, beams, or extruded domains. We consider both scenarios.

- The spatial domain Ω is partially separable. In this case the separated representation writes

$$u(\mathbf{x}, z, t) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot Z_i(z) \cdot T_i(t) \tag{100}$$

  where $\mathbf{x} = (x, y) \in \Omega_{xy}$, $z \in \Omega_z$ and $t \in \Omega_t$. Thus, iteration p of the alternated directions strategy at a given enrichment step n consists of

  1. Solving in Ω_{xy} a two-dimensional BVP to obtain function X_n^p,
  2. Solving in Ω_z a one-dimensional BVP to obtain function Z_n^p,
  3. Solving in Ω_t a one-dimensional IVP to obtain function T_n^p.

  The complexity of the PGD simulation scales with the two-dimensional mesh used to solve the BVPs in Ω_{xy}, regardless of the mesh and the time step used in the solution of the BVP and the IVPs defined in Ω_z and Ω_t for calculating functions $Z_i(z)$ and $T_i(t)$.
- The spatial domain Ω is fully separable. In this case the separated representation writes

$$u(x, y, z, t) = \sum_{i=1}^{N} X_i(x) \cdot Y_i(y) \cdot Z_i(z) \cdot T_i(t) \tag{101}$$

  Iteration p of the alternated directions strategy at a given enrichment step n consists of

  1. Solving in Ω_x a one-dimensional BVP to obtain the function X_n^p,
  2. Solving in Ω_y a one-dimensional BVP to obtain the function Y_n^p,
  3. Solving in Ω_z a one-dimensional BVP to obtain the function Z_n^p,
  4. Solving in Ω_t a one-dimensional IVP to obtain the function T_n^p.

  The cost savings provided by the PGD are potentially phenomenal when the spatial domain is fully separable. Indeed, the complexity of the PGD simulation now scales with the one-dimensional meshes used to solve the BVPs in Ω_x, Ω_y, and Ω_z, regardless of the time step used in the solution of the decoupled IVPs in Ω_t.

Even when the domain is not fully separable, a fully separated representation could be considered by using appropriate geometrical mappings or by immersing the nonseparable domain into a fully separable one. The interested reader can refer to Gonzalez *et al.* (2010) and Ghnatios *et al.* (2016).

In-plane–out-of-plane separated representations are particularly useful for addressing the solution of problems defined in plate (Bognet *et al.*, 2012), shell (Bognet *et al.*, 2014), or extruded domains (Leygue *et al.*, 2013). A parametric 3D elastic solution of beams involved in frame structures was proposed in Bordeu *et al.* (2015). The same approach was extensively considered in structural plate and shell models in Gallimard *et al.* (2013), Vidal *et al.* (2012, 2013, 2014a,b, 2015), and Pruliere (2014). Space-separated representations were enriched with discontinuous functions for representing cracks in Giner *et al.* (2013), delamination in Metoui *et al.* (2014), and thermal contact resistances in Chinesta *et al.* (2014). Domain decomposition within the separated space representation was accomplished in Nazeer *et al.* (2014), and localized behaviors were addressed by using the superposition techniques in Ammar *et al.* (2011b).

The in-plane–out-of-plane decomposition was then extended to many other physics problems: thermal models were considered in Chinesta *et al.* (2014); squeeze flows of Newtonian and non-Newtonian fluids in laminates in Ghnatios *et al.* (2015); flows in stratified porous media in Chinesta *et al.* (2011a), and nonlinear viscoplastic flows in plate domains in Canales *et al.* (2016). A full space decomposition was also efficiently applied for solving the Navier–Stokes equations in the lid-driven cavity problem in Dumon *et al.* (2011, 2013a,b).

4.5 Multidimensional models

Consider a problem defined in a high-dimensional space of dimension d for the unknown field $u(x_1, \dots, x_d)$. Here, the coordinates x_i denote any usual coordinates (scalar or vectorial) related to space, time, and/or any conformational coordinate.

We seek a solution for $u(x_1, \dots, x_d) \in \Omega_1 \times \cdots \times \Omega_d$. The PGD yields an approximate solution in the separated form:

$$u(x_1, \dots, x_d) \approx \sum_{i=1}^{N} X_i^1(x_1) \cdots X_i^d(x_d) = \sum_{i=1}^{N} \prod_{j=1}^{d} X_i^j(x_j) \tag{102}$$

If M nodes are used to discretize each coordinate, the total number of PGD unknowns is $N \cdot M \cdot d$ instead of the M^d degrees of freedom involved in standard mesh-based discretizations. Thus, the high-dimensional solution is computed by solving a number of low-dimensional problems, alleviating the so-called curse of dimensionality involved in high-dimensional models.

Separated representations within the PGD framework were applied for solving the multidimensional Fokker–Planck equation describing complex fluids within the kinetic theory framework in Ammar *et al.* (2006, 2007). The solution procedure was extended to nonlinear kinetic theory descriptions of more complex molecular models in Mokdad *et al.* (2007). In Leonenko and Phillips (2009), the authors considered multibead–spring models but used a spectral approximation for representing all the functions involved in the finite sums decomposition. A deeper analysis of nonlinear and transient models was considered in Ammar *et al.* (2010d). Complex fluid models were coupled with complex flows in Pruliere *et al.* (2009) and Mokdad *et al.* (2010), opening very encouraging perspectives and pointing out the necessity of defining efficient stabilizations. A first tentative of convective stabilization was proposed in Gonzalez *et al.* (2013). Finally, in Chinesta *et al.* (2007) the PGD was applied for solving the stochastic equation within the Brownian configuration field framework.

Multidimensional models encountered in the finer descriptions of matter (ranging from quantum chemistry to statistical mechanics descriptions) were revisited in Ammar *et al.* (2008). The multidimensional chemical master equation was solved in Ammar *et al.* (2012) and the Langer's equation governing phase transitions in Lamari *et al.* (2012).

4.6 Parametric solutions

This section illustrates how parameters of different nature become coordinates. The problems considered are quite simplistic, but the same rationale was considered for solving more complex problems reported at the end of the present section. We consider three types of parameters: (i) parameters related to the model; (ii) parameters related to the initial and boundary conditions; and (iii) geometrical parameters defining the space–time domain in which the model is defined.

4.6.1 Model parameters as extra coordinates

We consider the parametric heat transfer equation

$$\frac{\partial u}{\partial t} - k\Delta u - f = 0 \tag{103}$$

with homogeneous initial and boundary conditions. Here, $(\mathbf{x}, t, k) \in \Omega_x \times \Omega_t \times \Omega_k$. The scalar conductivity k is here viewed as a new coordinate defined in the interval Ω_k. Thus, instead of solving the thermal model for different discrete values of the conductivity parameter, we wish to solve only once a more general problem. For that purpose, we consider the weighted residual form related to equation (103):

$$\int_{\Omega \times \Omega_t \times \Omega_k} u^* \left(\frac{\partial u}{\partial t} - k\Delta u - f \right) \mathrm{d}\mathbf{x}\, \mathrm{d}t\, \mathrm{d}k = 0 \tag{104}$$

The PGD solution is sought in the form

$$u(\mathbf{x}, t, k) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \tag{105}$$

At iteration $n < N$, the solution $u^n(\mathbf{x}, t, k)$ reads

$$u^n(\mathbf{x}, t, k) = \sum_{i=1}^{n} X_i(\mathbf{x}) \cdot T_i(t) \cdot K_i(k) \tag{106}$$

and the new trial function $u^{n+1}(\mathbf{x}, t, k)$ is searched according to

$$u^{n+1}(\mathbf{x}, t, k) = u^n(\mathbf{x}, t, k) + X_{n+1}(\mathbf{x}) \cdot T_{n+1}(t) \cdot K_{n+1}(k) \tag{107}$$

with the test function $u^\star$ given by

$$u^\star(\mathbf{x},t,k) = X^\star(\mathbf{x})\cdot T_{n+1}(t)\cdot K_{n+1}(k) + X_{n+1}(\mathbf{x})\cdot T^\star(t) \cdot K_{n+1}(k) + X_{n+1}(\mathbf{x})\cdot T_{n+1}(t)\cdot K^\star(k) \quad (108)$$

By introducing the trial and test functions equations (107) and (108), respectively, into the weak form, equation (104), and using an appropriate linearization, functions $X_{n+1}(\mathbf{x})$, $T_{n+1}(t)$, and $K_{n+1}(k)$ are calculated. When considering the simplest linearization strategy, namely the alternated directions fixed point algorithm, the following steps are repeated until reaching convergence:

- With $T^{p-1}_{n+1}(t)$ and $K^{p-1}_{n+1}(k)$ given at the previous iteration of the nonlinear solver (arbitrarily initialized at the first iteration: $T^0_{n+1}(t)$ and $K^0_{n+1}(k)$), all the integrals in Ω_t and Ω_k are evaluated, leading to a BVP involving $X^p_{n+1}(\mathbf{x})$.
- With $X^p_{n+1}(\mathbf{x})$ just calculated and $K^{p-1}_{n+1}(k)$ given at the previous iteration of the nonlinear solver, all the integrals in $\Omega_x \times \Omega_k$ are evaluated, leading to a one-dimensional IVP involving $T^p_{n+1}(t)$.
- With $X^p_{n+1}(\mathbf{x})$ and $T^p_{n+1}(t)$ just updated, all the integrals in $\Omega_x \times \Omega_t$ are evaluated, leading to an algebraic problem involving $K^p_{n+1}(k)$.
- The convergence of the fixed point schema is checked.

4.6.2 Boundary conditions as extra coordinates

For the sake of simplicity, we first consider the steady-state heat equation

$$\nabla\cdot(\mathbf{K}\cdot\nabla u(\mathbf{x})) + f(\mathbf{x}) = 0 \quad (109)$$

with $\mathbf{x}\in\Omega\subset\mathbb{R}^3$, subjected to the boundary conditions

$$\begin{cases} u(\mathbf{x}\in\Gamma_d) = u_g \\ (-\mathbf{K}\cdot\nabla u)|_{\mathbf{x}\in\Gamma_n}\cdot\mathbf{n} = \mathbf{q}_g\cdot\mathbf{n} = q_g \end{cases} \quad (110)$$

with $\mathbf{K}$ the conductivity tensor and $\mathbf{n}$ the outward unit vector defined in the domain boundary Γ_n, with $\partial\Omega \equiv \Gamma = \Gamma_d \cup \Gamma_n$ and $\Gamma_d \cap \Gamma_n = \emptyset$.

In what follows, we consider the simplest scenario that consists of constant Neumann and Dirichlet boundary conditions. More complex and general situations were addressed in Chinesta *et al.* (2013b), where non-constant boundary and initial conditions were addressed.

- *Neumann boundary condition as extra coordinates.* First, imagine that we are interested in knowing the model solution for values of the heat flux $q_g \in \mathcal{I}_q = [q_g^-, q_g^+]$. We could consider the given heat flux as an extra-coordinate and then solve only once the resulting 4D heat equation for calculating the general parametric solution $u(\mathbf{x}, q_g)$. For this purpose, the solution is sought in the separated form

$$u(\mathbf{x},q_g) \approx \sum_{i=1}^{N} X_i(\mathbf{x})\cdot\mathcal{Q}_i(q_g) \quad (111)$$

In order to enforce the prescribed Dirichlet boundary condition $u(\mathbf{x}\in\Gamma_d) = u_g$, the simplest procedure consists of choosing the first functional couple $X_1(\mathbf{x})\cdot\mathcal{Q}_1(q_g)$ to ensure that $u^1(\mathbf{x}\in\Gamma_d, q_g) = X_1(\mathbf{x}\in\Gamma_d)\cdot\mathcal{Q}_1(q_g) = u_g$. Thus, the remaining terms of the finite sum $X_i(\mathbf{x})$, $i>1$, will be subjected to homogeneous essential boundary conditions, that is, $X_i(\mathbf{x}\in\Gamma_d) = 0$.
In order to use the approximation (111), we start by considering the weak form related to equation (109), which writes: Find $u(\mathbf{x})\in H^1(\Omega)$, verifying $u(\mathbf{x}\in\Gamma_d) = u_g$, such that

$$\int_\Omega \nabla u^*\cdot(\mathbf{K}\cdot\nabla u)\ \mathrm{d}\mathbf{x} = \int_{\Gamma_n} u^*\ (\mathbf{K}\cdot\nabla u)\cdot\mathbf{n}\ \mathrm{d}\mathbf{x} + \int_\Omega u^*\ f(\mathbf{x})\ \mathrm{d}\mathbf{x} \quad (112)$$

is verified $\forall u^*\in H^1(\Omega)$, with $u^*(\mathbf{x}\in\Gamma_d) = 0$.
Introducing the Neumann condition (110) into (112) results in

$$\int_\Omega \nabla u^*\cdot(\mathbf{K}\cdot\nabla u)\ \mathrm{d}\mathbf{x} = -\int_{\Gamma_n} u^*\ q_g\ \mathrm{d}\mathbf{x} + \int_\Omega u^* f(\mathbf{x})\ \mathrm{d}\mathbf{x} \quad (113)$$

For using the approximation (111), we must consider the extended-weak form defined in the domain $\Omega\times\mathcal{I}_q$:

$$\int_{\Omega\times\mathcal{I}_q} \nabla u^*\cdot(\mathbf{K}\cdot\nabla u)\ \mathrm{d}\mathbf{x}\ \mathrm{d}q_g = -\int_{\Gamma_n\times\mathcal{I}_q} u^*\ q_g\ \mathrm{d}\mathbf{x}\ \mathrm{d}q_g + \int_{\Omega\times\mathcal{I}_q} u^* f(\mathbf{x})\ \mathrm{d}\mathbf{x}\ \mathrm{d}q_g \quad (114)$$

By assuming at iteration n

$$\begin{cases} u^n(\mathbf{x},q_g) = \sum_{i=1}^{n-1} X_i(\mathbf{x})\cdot\mathcal{Q}_i(q_g) + X_n(\mathbf{x})\cdot\mathcal{Q}_n(q_g) \\ \qquad\qquad = u^{n-1}(\mathbf{x},q_g) + X_n(\mathbf{x})\cdot\mathcal{Q}_n(q_g) \\ u^* = X^*(\mathbf{x})\cdot\mathcal{Q}_n(q_g) + X_n(\mathbf{x})\cdot\mathcal{Q}^*(q_g) \end{cases} \quad (115)$$

Now, the double iteration described in the previous section – one for enriching the separated representation and the second one for solving the nonlinear problem arising at each enrichment iteration – is performed in order to calculate the solution-separated representation.

- *Dirichlet boundary condition as extra coordinate.* Now, assume that we are interested in considering the solution of model (109) for any value of u_g in (110) in a certain interval $\mathcal{I}_u = [u_g^-, u_g^+]$. For this purpose, we consider the function $\varphi(\mathbf{x})$ continuous in $\overline{\Omega}$ such that $\Delta\varphi \in L_2(\Omega)$ and $\varphi(\mathbf{x} \in \Gamma_d) = 1$. Thus, we can define the change of variable (Gonzalez *et al.*, 2010)

$$u(\mathbf{x}) = v(\mathbf{x}) + u_g \ \varphi(\mathbf{x}) \tag{116}$$

which allows rewriting equations (109) and (110) as

$$\nabla \cdot (\mathbf{K} \cdot \nabla v(\mathbf{x})) + u_g \ \nabla \cdot (\mathbf{K} \cdot \nabla \varphi(\mathbf{x})) + f(\mathbf{x}) = 0 \tag{117}$$

subjected to the boundary conditions

$$\begin{cases} v(\mathbf{x} \in \Gamma_d) = 0 \\ (-\mathbf{K} \cdot \nabla v)|_{\mathbf{x}\in\Gamma_n} \cdot \mathbf{n} = u_g \ (\mathbf{K} \cdot \nabla \varphi)|_{\mathbf{x}\in\Gamma_n} \cdot \mathbf{n} + q_g \end{cases} \tag{118}$$

which results in the weak form

$$\begin{aligned} \int_\Omega \nabla v^* \cdot (\mathbf{K} \cdot \nabla v) \ \mathrm{d}\mathbf{x} &= -\int_\Omega u_g \ \nabla v^* \cdot (\mathbf{K} \cdot \nabla \varphi) \ \mathrm{d}\mathbf{x} \\ &+ \int_\Omega v^* \ f(\mathbf{x}) \ \mathrm{d}\mathbf{x} - \int_{\Gamma_n} v^* \ q_g \ \mathrm{d}\mathbf{x} \\ &- \int_{\Gamma_n} v^* \ u_g \ (\mathbf{K} \cdot \nabla \varphi) \cdot \mathbf{n} \ \mathrm{d}\mathbf{x} \end{aligned} \tag{119}$$

We can now introduce u_g as the extra coordinate, searching for the solution in the separated form:

$$v(\mathbf{x}, u_g) \approx \sum_{i=1}^{N} X_i(\mathbf{x}) \cdot \mathcal{U}_i(u_g) \tag{120}$$

which needs for the extended weak form

$$\begin{aligned} &\int_{\Omega\times\mathcal{I}_u} \nabla v^* \cdot (\mathbf{K} \cdot \nabla v) \ \mathrm{d}\mathbf{x} \ \mathrm{d}u_g = \\ &- \int_{\Omega\times\mathcal{I}_u} u_g \ \nabla v^* \cdot (\mathbf{K} \cdot \nabla \varphi) \ \mathrm{d}\mathbf{x} \ \mathrm{d}u_g \\ &+ \int_{\Omega\times\mathcal{I}_u} v^* f(\mathbf{x}) \ \mathrm{d}\mathbf{x} \ \mathrm{d}u_g - \int_{\Gamma_n\times\mathcal{I}_u} v^* \ q_g \ \mathrm{d}\mathbf{x} \ \mathrm{d}u_g \\ &- \int_{\Gamma_n\times\mathcal{I}_u} v^* \ u_g \ (\mathbf{K} \cdot \nabla \varphi) \cdot \mathbf{n} \ \mathrm{d}\mathbf{x} \ \mathrm{d}u_g \end{aligned} \tag{121}$$

on which the alternated directions fixed point algorithm is applied again to calculate the parametric solution (120).

4.6.3 Parametric domains

For the sake of clarity and without loss of generality, we are addressing in this section the transient one-dimensional heat equation

$$\frac{\partial u}{\partial t} = k\frac{\partial^2 u}{\partial x^2} + f \tag{122}$$

with $t \in \Omega_t = (0, \Theta]$, $x \in \Omega_x = (0, L)$, constant conductivity k and source term f, and homogeneous initial and boundary conditions, that is, $u(x = 0, t) = u(x = L, t) = u(x, t = 0) = 0$.

The associated space–time weak form reads

$$\begin{aligned} \int_{\Omega_x\times\Omega_t} u^* \ \frac{\partial u}{\partial t} \ \mathrm{d}x \ \mathrm{d}t &= -\int_{\Omega_x\times\Omega_t} k\frac{\partial u^*}{\partial x} \ \frac{\partial u}{\partial x} \ \mathrm{d}x \ \mathrm{d}t \\ &+ \int_{\Omega_x\times\Omega_t} u^* \ f \ \mathrm{d}x \ \mathrm{d}t \end{aligned} \tag{123}$$

If we are interested in computing the solution $u(x, t)$ in many domains of length $L \in \Omega_L = [L^-, L^+]$ and for many time intervals of length $\Theta \in \Omega_\Theta = [\Theta^-, \Theta^+]$. More than solving the model for many possible choice in order to define a meta-model, it is preferable to compute the parametric solution by considering L and Θ as extra coordinates. However, equation (123) does not involve an explicit dependence on the extra coordinates L and Θ, both defining the domain of integration. In order to make explicit this dependence, we consider the coordinate transformation

$$\begin{cases} t = \tau \ \Theta, \ \tau \in \mathcal{I} = [0, 1] \\ x = \lambda \ L, \ \lambda \in \mathcal{I} = [0, 1] \end{cases} \tag{124}$$

In this case, the weak form (123) reads

$$\begin{aligned} \int_{\mathcal{I}\times\mathcal{I}} u^*\frac{\partial u}{\partial \tau} L \ \mathrm{d}\lambda \ \mathrm{d}\tau &= -\int_{\mathcal{I}\times\mathcal{I}} k\frac{\partial u^*}{\partial \lambda}\frac{\partial u}{\partial \lambda}\frac{\Theta}{L} \ \mathrm{d}\lambda \ \mathrm{d}\tau \\ &+ \int_{\mathcal{I}\times\mathcal{I}} u^* f L\Theta \ \mathrm{d}\lambda \ \mathrm{d}\tau \end{aligned} \tag{125}$$

which allows calculating the parametric solution $u(\tau, \lambda, L, \Theta)$ by considering the separated representation

$$u(\lambda, \tau, L, \Theta) \approx \sum_{i=1}^{N} X_i(\lambda) \cdot T_i(\tau) \cdot \mathcal{L}_i(L) \cdot \mathcal{T}_i(\Theta) \tag{126}$$

and the extended weak form

$$\int_{\mathcal{I}\times\mathcal{I}\times\Omega_L\times\Omega_\Theta} u^* \frac{\partial u}{\partial \tau} L \, \mathrm{d}\lambda \, \mathrm{d}\tau \, \mathrm{d}L \, \mathrm{d}\Theta =$$
$$- \int_{\mathcal{I}\times\mathcal{I}\times\Omega_L\times\Omega_\Theta} k \frac{\partial u^*}{\partial \lambda} \frac{\partial u}{\partial \lambda} \frac{\Theta}{L} \, \mathrm{d}\lambda \, \mathrm{d}\tau \, \mathrm{d}L \, \mathrm{d}\Theta$$
$$+ \int_{\mathcal{I}\times\mathcal{I}\times\Omega_L\times\Omega_\Theta} u^* f L \Theta \, \mathrm{d}\lambda \, \mathrm{d}\tau \, \mathrm{d}L \, \mathrm{d}\Theta \qquad (127)$$

4.6.4 Related works

This kind of parametric modeling was widely addressed in a panoply of applications, where material parameters (Pruliere *et al.*, 2010; Ammar *et al.*, 2010c; Lamari *et al.*, 2010; Bognet *et al.*, 2012; Aghighi *et al.*, 2015; Beringhier and Gigliotti, 2015), initial conditions (Gonzalez *et al.*, 2012, 2014), boundary conditions (Ghnatios *et al.*, 2011, 2012; Niroomandi *et al.*, 2013; Gonzalez *et al.*, 2015), and parameters defining the geometry (Leygue and Verron, 2010; Nouy, 2011; Ammar *et al.*, 2014; Zlotnik *et al.*, 2015) were considered extra coordinates within the PGD framework. All these parametric solutions were successfully employed for performing real-time simulations (e.g., surgical simulation involving haptic devices addressing contact, cutting, etc.), material homogenization, real-time process optimization, inverse analysis, and simulation-based control. They where also employed in DDDASs. Vitse *et al.* (2014) adopted the just-referred espace–time parameter-separated representation for constructing parametric solutions.

For the treatment of the nonlinearities involved in the works just referred, the separated representation constructors were combined with numerous nonlinear solvers ranging from the most standard ones (fixed point, Newton, etc.) to less standard approaches based on the LATIN, asymptotic numerical method (e.g., Niroomandi *et al.*, 2013; Leygue *et al.*, 2013, among many others), or the discrete empirical interpolation method (Chinesta *et al.*, 2013b).

In the context of stochastic modeling, PGD was introduced in Nouy (2007) for the uncertainty quantification and propagation. The interpretation of the separated representation constructor as a generalized eigenproblem allowed defining dedicated algorithms inspired from solution techniques for classical eigenproblems (Nouy, 2008). In this context, deterministic and stochastic contributions were separated, making the PGD a promising alternative to traditional methods for uncertainty propagation, as discussed in Nouy and Le Maitre (2009). PGD has also been extended to stochastic nonlinear problems in Nouy (2009). More recently, PGD has been successfully applied to the solution of high-dimensional stochastic parametric problems, with the introduction of suitable hierarchical tensor representations and associated algorithms (Nouy, 2010b).

4.7 Concluding remarks

Even though noticeable progress has been accomplished in PGD concerning its mathematical foundations, the numerical technology, and its applications, many issues persist and are being deeply investigated. Among them, we can cite the robust and efficient separated representation constructor (first attempts in this direction were considered in Nouy, 2010a; Billaud-Friess *et al.*, 2014), stabilized formulation implying advection or mixed formulation, nonlinear multiparametric problems (strongly nonlinear and involving tens or hundreds of parameters), parametric models defined in evolving domains, localized behaviours, and the numerical analysis of nonsymmetric and nonlinear operators.

5 CONCLUSIONS

In this chapter, we revisited the state of the art and the most recent developments of three families of model reduction techniques: POD, RB, and PGD. It highlighted the exponential growth in recent years of several scientific contributions dealing with the broad topic, and also the gradual transfer from academic problems to the ones of interest in science and technology. Model reduction techniques have nowadays become an appealing route for solving more and more complex models but in a fast and cheap way, many times and in real time by making use of deployed computing platforms. These features allow democratizing numerical simulations, which can no longer be restricted to big research centers of industries and universities, thus promoting and speeding up innovation.

Despite the significant progress accomplished in recent years, issues still remain, requiring intensive work and new ideas for addressing more and more complex models, circumventing the remaining issues described in this chapter and demonstrating the robustness of the applied methodologies.

6 RELATED CHAPTERS

(*See also* **Finite Difference Methods**, **Finite Element Methods**, **Adaptive Computational Methods for Parabolic Problems**, **Uncertainty Quantification and Bayesian Inversion**)

ACKNOWLEDGMENTS

F. Chinesta acknowledges the contributions of his group at ECN, of E. Cueto and A. Ammar and their groups as well as the funding support from ESI GROUP.

G. Rozza acknowledges support provide by European Research Council (Project ERC CoG 2015 AROMA-CFD GA 681447) and crucial inputs from Prof. A.T. Patera (MIT).

K. Willcox acknowledges helpful input from B. Peherstorfer and funding support from the the Dynamic Data-Driven Application Systems Program (program manager F. Darema) under AFOSR grant FA9550-16-1-0108.

NOTES

1. http://dddas.org/.
2. Typically, the different subdomains correspond to different materials and hence material properties, or more generally different (discontinuously varying in space) PDE coefficients; however, the subdomains may also be introduced for algorithmic purposes to ensure well-behaved mappings.
3. As regards smoothness, we note that for $\Theta^q \in C^\infty(\mathcal{D})$, $1 \le q \le Q$, it can be shown under our coercivity, continuity, and affine hypotheses of Section 3.2 that $\|D^\sigma u^{N_h}(\boldsymbol{\mu})\|_X$ is bounded by a constant $C^{|\sigma|}$ (independent of N_h) for all $\boldsymbol{\mu} \in \mathcal{D}$; here $D^\sigma u^{N_h}(\boldsymbol{\mu})$ refers to the σ multi-index derivative of u^{N_h} with respect to $\boldsymbol{\mu}$.
4. It thus follows that the *a posteriori* error estimation contribution to the cost of the greedy algorithm of Section 3.6 is $O(QN_{\max}N_h^{\cdot}) + O(Q^2N_{\max}^2N_h) + O(n_{\text{train}}Q^2N_{\max}^3)$: we may thus choose N_h and n_{train} *independently* (and large).

REFERENCES

Aghighi MS, Ammar A, Metivier C and Chinesta F. Parametric solution of the Rayleigh-Bnard convection model by using the PGD: application to nanofluids. *Int. J. Numer. Methods Heat Fluid Flow* 2015; **25**(6):1252–1281.

Aghighi MS, Ammar A, Metivier C, Normandin M and Chinesta F. Non incremental transient solution of the Rayleigh-Bnard convection model using the PGD. *J. Non-Newtonian Fluid Mech.* 2013; **200**:65–78.

Aguado JV, Huerta A, Chinesta F and Cueto E. Real-time monitoring of thermal processes by reduced order modelling. *Int. J. Numer. Methods Eng.* 2015; **102**(5):991–1017.

Almroth BO, Stern P and Brogan FA. Automatic choice of global shape functions in structural analysis. *AIAA J.* 1978; **16**:525–528.

Ammar A, Chinesta F and Cueto E. Coupling finite elements and proper generalized decompositions. *Int. J. Multiscale Comput. Eng.* 2011a; **9**(1):17–33.

Ammar A, Chinesta F, Cueto E and Doblare M. Proper generalized decomposition of time-multiscale models. *Int. J. Numer. Methods Eng.* 2011b; **90**(5):569–596.

Ammar A, Chinesta F and Joyot P. The nanometric and micrometric scales of the structure and mechanics of materials revisited: an introduction to the challenges of fully deterministic numerical descriptions. *Int. J. Multiscale Comput. Eng.* 2008; **6**(3):191–213.

Ammar A, Chinesta F, Diez P and Huerta A. An error estimator for separated representations of highly multidimensional models. *Comput. Methods Appl. Mech. Eng.* 2010a; **199**:1872–1880.

Ammar A, Chinesta F and Falco A. On the convergence of a greedy rank-one update algorithm for a class of linear systems. *Arch. Comput. Methods Eng.* 2010b; **17**(4):473–486.

Ammar A, Normandin M and Chinesta F. Solving parametric complex fluids models in rheometric flows. *J. Non-Newtonian Fluid Mech.* 2010c; **165**:1588–1601.

Ammar A, Normandin M, Daim F, Gonzalez D, Cueto E and Chinesta F. Non-incremental strategies based on separated representations: applications in computational rheology. *Commun. Math. Sci.* 2010d; **8**(3):671–695.

Ammar A, Cueto E and Chinesta F. Reduction of the chemical master equation for gene regulatory networks using proper generalized decompositions. *Int. J. Numer. Methods Biomed. Eng.* 2012; **28**(9):960–973.

Ammar A, Huerta A, Chinesta F, Cueto E and Leygue A. Parametric solutions involving geometry: a step towards efficient shape optimization. *Comput. Methods Appl. Mech. Eng.* 2014; **268C**:178–193.

Ammar A, Mokdad B, Chinesta F and Keunings R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *J. Non-Newtonian Fluid Mech.* 2006; **139**:153–176.

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

Ammar A, Zghal A, Morel F and Chinesta F. On the space-time separated representation of integral linear viscoelastic models. *C.R. Mcanique* 2015; **343**(4):247–263.

Amsallem D and Farhat C. Interpolation method for the adaptation of reduced-order models to parameter changes and its application to aeroelasticity. *AIAA J.* 2008; **46**(10):1803–1813.

Antoulas A. *Approximation of Large-Scale Dynamical Systems*. SIAM Publications: Philadelphia, PA, 2005.

Antoulas A, Beattie C and Gugercin S. Interpolatory model reduction of large-scale dynamical systems. In *Efficient Modeling and Control of Large-Scale Systems*, Mohammadpour J and Grigoriadis K (eds). Springer-Verlag, 2010; 2–58.

Astrid P, Weiland S, Willcox K and Backx T. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Trans. Autom. Control* 2008; **53**(10):2237–2251.

Bai Z. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Appl. Numer. Math.* 2002; **43**(1–2):9–44.

Barbarulo A, Ladeveze P, Riou H and Kovalevsky L. Proper generalized decomposition applied to linear acoustic: a new tool for broad band calculation. *J. Sound Vib.* 2014a; **333**(11):2422–2431.

Barbarulo A, Riou H, Kovalevsky L and Ladeveze P. PGD-VTCR: a reduced order model technique to solve medium frequency broad band problems on complex acoustical systems. *J. Mech. Eng.* 2014b; **60**(5):307–314.

Barrault M, Nguyen NC, Maday Y and Patera AT. An "empirical interpolation" method: application to efficient reduced-basis discretization of partial differential equations. *C.R. Acad. Sci. Paris, Sér. I* 2004; **339**:667–672.

Barrett A and Reddien G. On the reduced basis method. *Z. Angew. Math. Mech.* 1995; **75**(7):543–549.

Baur U, Benner P, Beattie C and Gugercin S. Interpolatory projection methods for parameterized model reduction. *SIAM J. Sci. Comput.* 2011; **33**:2489–2518.

Benner P, Gugercin S and Willcox K. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Rev.* 2015; **57**(4):483–531.

Beringhier M and Gigliotti M. A novel methodology for the rapid identification of the water diffusion coefficients of composite materials. *Composites Part A* 2015;**68**:212–218.

Bernoulli Ch. *Vademecum des Mechanikers*. Cotta: Stuttgart, 1836.

Billaud-Friess M, Nouy A and Zahm O. A tensor approximation method based on ideal minimal residual formulations for the solution of high-dimensional problems. *Math. Modell. Numer. Anal.* 2014; **48**:1777–1806.

Blanze C, Champaney L, Cognard J-Y and Ladeveze P. A modular approach to structure assembly computations - application to contact problems. *Eng. Comput.* 1996; **13**(1):15.

Bognet B, Leygue A and Chinesta F. Separated representations of 3D elastic solutions in shell geometries. *Adv. Modell. Simul. Eng. Sci.* 2014; **1**:4, http://www.amses-journal.com/content/1/1/4.

Bognet B, Leygue A, Chinesta F, Poitou A and Bordeu F. Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. *Comput. Methods Appl. Mech. Eng.* 2012; **201**:1–12.

Bonithon G, Joyot P, Chinesta F and Villon P. Non-incremental boundary element discretization of parabolic models based on the use of Proper Generalized Decompositions. *Eng. Anal. Bound. Elem.* 2011; **35**(1):2–17.

Bordeu F, Ghnatios Ch, Boulze D, Carles B, Sireude D, Leygue A and Chinesta F. Parametric 3D elastic solutions of beams involved in frame structures. *Adv. Aircr. Spacecr. Sci.* 2015; **2**(3):233–248.

Boucard PA and Ladeveze P. A multiple solution method for non-linear structural mechanics. *Mech. Eng.* 1999; **50**(5):317–328.

Boucard PA, Ladeveze P, Poss M and Rougée P. A non-incremental approach for large displacement problems. *Comput. Struct.* 1997; **64**:499–508.

Boucinha L, Ammar A, Gravouil A and Nouy A. Ideal minimal residual-based proper generalized decomposition for non-symmetric multi-field models - application to transient elastodynamics in space-time domain. *Comput. Methods Appl. Mech. Eng.* 2014; **273**:56–76.

Boucinha L, Gravouil A and Ammar A. Space-time proper generalized decompositions for the resolution of transient elastodynamic models. *Comput. Methods Appl. Mech. Eng.* 2013; **255**:67–88.

Brand M. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra Appl.* 2006; **415**(1):20–30.

Buffa A, Maday Y, Patera AT, Prud'homme C and Turinici G. *A priori* convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM Math. Modell. Numer. Anal.* 2012; **46**:595–603.

Bui-Thanh T, Damodaran M and Willcox K. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *Proceedings of the 21st Applied Aerodynamics AIAA Conference*, Orlando, FL, 2003, AIAA Paper 2003-4213.

Bui-Thanh T, Damodaran M and Willcox K. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA J.* 2004; **42**(8):1505–1516.

Bui-Thanh T, Willcox K and Ghattas O. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM J. Sci. Comput.* 2008; **30**(6):3270–3288.

Canales D, Leygue A, Chinesta F, Alfaro I, Gonzalez D, Cueto E, Feulvarch E and Bergheau JM. In-plane-out-of-plane separated representations of updated-Lagrangian descriptions of thermomechanical models defined in plate domains. *C.R. Acad. Sci.* 2016; **344**(4-5):225–235.

Cancès E, Defranceschi M, Kutzelnigg W, Le Bris C and Maday Y. *Computational Quantum Chemistry: A Primer, Handbook of Numerical Analysis*, vol. **X**. Elsevier: North-Holland, Amsterdam, 2003; 3–270.

Cancès E, Ehrlacher V and Lelièvre T. Convergence of a greedy algorithm for high-dimensional convex nonlinear problems. *Math. Models Methods Appl. Sci.* 2011; **21**:2433–2468.

Cancès E, Ehrlacher V and Lelièvre T. Greedy algorithms for high-dimensional non-symmetric linear problems. *ESAIM Proc.* 2013; **41**:95–131.

Canuto C, Tonn T and Urban K. A posteriori error analysis of the reduced basis method for nonaffine parametrized nonlinear PDEs. *SIAM J. Numer. Anal.* 2009; **47**:2001–2022.

Carlberg K, Farhat C, Cortial J and Amsallem D. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *J. Comput. Phys.* 2013; **242**:623–647.

Champaney L, Cognard J-Y, Dureisseix D and Ladeveze P. Large scale applications on parallel computers of a mixed domain decomposition method. *Comput. Mech.* 1997; **19**(4):253–263.

Chaturantabut S and Sorensen D. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* 2010; **32**(5):2737–2764.

Chen P, Quarteroni A and Rozza G. Comparison between reduced basis and stochastic collocation methods for elliptic problems. *J. Sci. Comput.* 2014; **59**:187–216.

Chinesta F, Ammar A and Cueto E. Proper generalized decomposition of multiscale models. *Int. J. Numer. Methods Eng.* 2010a; **83**(8-9):1114–1132.

Chinesta F, Ammar A and Cueto E. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Arch. Comput. Methods Eng.* 2010b; **17**(4):327–350.

Chinesta F, Ammar A, Falco A and Laso M. On the reduction of stochastic kinetic theory models of complex fluids. *Model. Simul. Mater. Sci. Eng.* 2007; **15**:639–652.

Chinesta F, Ammar A, Leygue A and Keunings R. An overview of the proper generalized decomposition with applications in computational rheology. *J. Non-Newtonian Fluid Mech.* 2011a; **166**:578–592.

Chinesta F, Ladeveze P and Cueto E. A short review in model order reduction based on proper generalized decomposition. *Arch. Comput. Methods Eng.* 2011b; **18**:395–404.

Chinesta F, Keunings R and Leygue A. *The Proper Generalized Decomposition for Advanced Numerical Simulations. A Primer*, Springerbriefs. Springer Cham: Heidelberg, New York, Dordrecht London, 2013a.

Chinesta F, Leygue A, Bordeu F, Aguado JV, Cueto E, Gonzalez D, Alfaro I, Ammar A and Huerta A. Parametric PGD based computational vademecum for efficient design, optimization and control. *Arch. Comput. Methods Eng.* 2013b; **20**(1):31–59.

Chinesta F, Leygue A, Bognet B, Ghnatios Ch, Poulhaon F, Bordeu F, Barasinski A, Poitou A, Chatel S and Maison-Le-Poec S. First steps towards an advanced simulation of composites manufacturing by automated tape placement. *Int. J. Mater. Form.* 2014; **7**(1):81–92.

Cremonesi M, Neron D, Guidault PA and Ladeveze P. A PGD-based homogenization technique for the resolution of nonlinear multiscale problems. *Comput. Methods Appl. Mech. Eng.* 2013; **267**:275–292.

Deparis S and Rozza G. Reduced basis method for multi-parameter-dependent steady Navier–Stokes equations: applications to natural convection in a cavity. *J. Comput. Phys.* 2009; **228**:4359–4378.

DeVore R, Petrova G and Wojtaszczyk P. Greedy algorithms for reduced bases in banach spaces. *Constr. Approx.* 2013; **37**:455–466.

DeVore RA and Temlyakov VN. Some remarks on greedy algorithms. *Adv. Comput. Math.* 1996; **5**:173–187.

Dowell E and Hall K. Modeling of fluid-structure interaction. *Annu. Rev. Fluid Mech.* 2001; **33**:445–490.

Dumon A, Allery C and Ammar A. Proper general decomposition (PGD) for the resolution of Navier-Stokes equations. *J. Comput. Phys.* 2011; **230**(4):1387–1407.

Dumon A, Allery C and Ammar A. Proper generalized decomposition method for incompressible Navier-Stokes equations with a spectral discretization. *Appl. Math. Comput.* 2013a; **219**(15):8145–8162.

Dumon A, Allery C and Ammar A. Simulation of heat and mass transport in a square lid-driven cavity with proper generalized decomposition. *Numer. Heat Transfer, Part B* 2013b; **63**(1):18–43.

Dureisseix D, Ladeveze P, Neron D and Schrefler B. A multi-time-scale strategy for multiphysics problems: application to poroelasticity. *Int. J. Multiscale Comput. Eng.* 2003a; **1**(4):387–400.

Dureisseix D, Ladeveze P and Schrefler B. A latin computational strategy for multiphysics problems: application to poroelasticity. *Int. J. Numer. Methods Eng.* 2003b; **56**(10):1489–1510.

Ehrlacher V. Convergence results on greedy algorithms for high-dimensional eigenvalue problems. *ESAIM, Proc. Surv.* 2014; **45**:148–157.

Everson R and Sirovich L. The Karhunen-Loeve procedure for gappy data. *J. Opt. Soc. Am.* 1995; **12**:1657–1664.

Falcó A and Nouy A. A proper generalized decomposition for the solution of elliptic problems in abstract form by using a functional Eckart-Young approach. *J. Math. Anal. Appl.* 2011a; **376**:469–480.

Falco A and Nouy A. Proper generalized decomposition for nonlinear convex problems in tensor Banach spaces. *Numer. Math.* 2011b; **121**(3):503–530.

Falco A, Hilario L, Monts N and Mora MC. Numerical strategies for the Galerkin-Proper generalized decomposition method. *Math. Comput. Modell.* 2013; **57**(78):1694–1702.

Figueroa L and Suli E. Greedy approximation of high-dimensional Ornstein-Uhlenbeck operators with unbounded drift, 2011. arXiv: http://arxiv.org/abs/1103.0726.

Fink JP and Rheinboldt WC. On the error behavior of the reduced basis technique for nonlinear finite element approximations. *Z. Angew. Math. Mech.* 1983; **63**(1):21–28.

Fox RL and Miura H. An approximate analysis technique for design calculations. *AIAA J.* 1971; **9**(1):177–179.

Freund R. Model reduction methods based on Krylov subspaces. *Acta Numer.* 2003; **12**:267–319.

Gallimard L, Vidal P and Polit O. Coupling finite element and reliability analysis through proper generalized decomposition model reduction. *Int. J. Numer. Methods Eng.* 2013; **95**(13):1079–1093.

Germoso C, Aguado JV, Fraile A, Alarcon E and Chinesta F. Efficient PGD-based dynamic calculation of non-linear soil behavior. *C.R. Acad. Sci.* 2016; **344**(1):24–41.

Gerner A-L and Veroy K. Certified reduced basis methods for parametrized saddle point problems. *SIAM J. Sci. Comput.* 2012; **34**:A2812–A2836.

Ghnatios Ch, Chinesta F and Binetruy Ch. The squeeze flow of composite laminates. *Int. J. Mater. Form.* 2015; **8**:73–83.

Ghnatios Ch, Chinesta F, Cueto E, Leygue A, Breitkopf P and Villon P. Methodological approach to efficient modeling and optimization of thermal processes taking place in a die: application to pultrusion. *Composites Part A* 2011; **42**:1169–1178.

Ghnatios Ch, Masson F, Huerta A, Cueto E, Leygue A and Chinesta F. Proper generalized decomposition based dynamic data-driven control of thermal processes. *Comput. Methods Appl. Mech. Eng.* 2012; **213**:29–41.

Ghnatios Ch, Xu G, Visonneau M, Leygue A, Chinesta F and Cimetiere A. On the space separated representation when addressing the solution of PDE in complex domains. *Discrete Contin. Dyn. Syst.* 2016; **9**(2):475–500.

Giacoma A, Dureisseix D, Gravouil A and Rochette M. Toward an optimal a priori reduced basis strategy for frictional contact problems with LATIN solver. *Comput. Methods Appl. Mech. Eng.* 2015; **283**:1357–1381.

Giner E, Bognet B, Rodenas JJ, Leygue A, Fuenmayor J and Chinesta F. The proper generalized decomposition (PGD) as a numerical procedure to solve 3D cracked plates in linear elastic fracture mechanics. *Int. J. Solid Struct.* 2013; **50**(10):1710–1720.

Gonzalez D, Alfaro I, Quesada C, Cueto E and Chinesta F. Computational vademecums for the real-time simulation of haptic collision between nonlinear solids. *Comput. Methods Appl. Mech. Eng.* 2015; **283**:210–223.

Gonzalez D, Ammar A, Chinesta F and Cueto E. Recent advances in the use of separated representations. *Int. J. Numer. Methods Eng.* 2010; **81**(5):637–659.

Gonzalez D, Cueto E and Chinesta F. Real-time direct integration of reduced solid dynamics equations. *Int. J. Numer. Methods Eng.* 2014; **99**(9):633–653.

Gonzalez D, Cueto E, Chinesta F, Diez P and Huerta A. SUPG-based stabilization of proper generalized decompositions for high-dimensional advection-diffusion equations. *Int. J. Numer. Methods Eng.* 2013; **94**(13):1216–1232.

Gonzalez D, Masson F, Poulhaon F, Leygue A, Cueto E and Chinesta F. Proper generalized decomposition based dynamic data-driven inverse identification. *Math. Comput. Simul.* 2012; **82**(9):1677–1695.

Grepl M, Maday Y, Nguyen NC and Patera AT. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *M2AN-Math. Modell. Numer. Anal.* 2007; **41**(3):575–605.

Grepl M and Patera AT. *A Posteriori* error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *M2AN Math. Modell. Numer. Anal.* 2005; **39**(1):157–181.

Haasdonk B and Ohlberger M. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Math. Modell. Numer. Anal.* 2008; **42**:277–302.

Hammoud M, Beringhier M and Grandidier JC. A reduced simulation applied to the viscoelastic fatigue of polymers. *C.R. Mcanique* 2014; **342**(12):671–691.

Hay A, Borggaard J and Pelletier D. Local improvements to reduced-order models using sensitivity analysis of the proper orthogonal decomposition. *J. Fluid Mech.* 2009; **629**:41–72.

Henneron T and Clenet S. Model order reduction of quasi-static problems based on POD and PGD approaches. *Eur. Phys. J. Appl. Phys.* 2013; **64**(2):24514.

Henneron T and Clenet S. Proper generalized decomposition method applied to solve 3-D magnetoquasi-static field problems coupling with external electric circuits. *IEEE Trans. Magn.* 2015; **51**(6):7208910.

Hesthaven J, Rozza G and Stamm B. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, Springer Briefs in Mathematics. Springer-Verlag, Cham, 2015.

Heyberger Ch, Boucard PA and Neron D. A rational strategy for the resolution of parametrized problems in the PGD framework. *Comput. Methods Appl. Mech. Eng.* 2013; **259**:40–49.

Hinze M and Volkwein S. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: error estimates and suboptimal control. In *Dimension Reduction of Large-Scale Systems*, *Lecture Notes in Computational and Applied Mathematics*, vol. **45**, Benner P, Mehrmann V and Sorensen D (eds). Springer: Berlin, Heidelberg, 2005; 261–306.

Hinze M and Volkwein S. Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition. *Comput. Optim. Appl.* 2008; **39**(3):319–345.

Hotelling H. Analysis of a complex of statistical variables with principal components. *J. Educ. Psychol.* 1933; **24**:417–441, 498–520.

Huynh DBP, Rozza G, Sen S and Patera AT. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *C.R. Acad. Sci. Paris, Anal. Numér., Ser. I* 2007; **345**:473–478.

Ito K and Ravindran SS. A reduced-order method for simulation and control of fluid flows. *J. Comput. Phys.* 1998; **143**(2):403–425.

Kim T. Frequency-domain Karhunen-Loeve method and its application to linear dynamic systems. *AIAA J.* 1998; **36**(11):2117–2123.

Kosambi D. Statistics in function space. *J. Indian Math. Soc.* 1943; **7**:76–88.

Kunisch K and Volkwein S. Control of Burgers' equation by reduced order approach using proper orthogonal decomposition. *J. Optim. Theory Appl.* 1999; **102**:345–371.

Kunisch K and Volkwein S. Optimal snapshot location for computing POD basis functions. *ESAIM: Math. Modell. Numer. Anal.* 2010; **44**(03):509–529.

Ladeveze P. On a family of algorithms for structural mechanics. *C.R. Acad. Sci. Paris* 1985; **300**(2):41–44 (in french).

Ladeveze P. The large time increment method for the analyze of structures with nonlinear constitutive relation described by internal variables. *C.R. Acad. Sci. Paris* 1989; **309**:1095–1099.

Ladeveze P. *Mécanique Non linéaire des Structures*. Hermès: Paris, 1996.

Ladeveze P. *Nonlinear Computationnal Structural Mechanics*, New Approaches and Non-Incremental Methods of Calculation. Springer-Verlag: New York, 1999.

Ladeveze P and Chamoin L. On the verification of model reduction methods based on the proper generalized decomposition. *Comput. Methods Appl. Mech. Eng.* 2011; **200**:2032–2047.

Ladeveze P and Dureisseix D. A 2-level and mixed domain decomposition approach for structural analysis. *Contemp. Math.* 1998; **218**:246–253.

Ladeveze P, Neron D and Gosselet P. On a mixed and multiscale domain decomposition method. *Comput. Methods Appl. Mech. Eng.* 2007; **96**:1526–1540.

Ladeveze P and Nouy A. On a multiscale computational strategy with time and space homogenization for structural mechanics. *Comput. Methods Appl. Mech. Eng.* 2003; **192**(28–30):3061–3087.

Ladeveze P, Passieux J-C, Neron D. The latin multiscale computational method and the proper generalized decomposition. *Comput. Methods Appl. Mech. Eng.* 2010; **199**(21-22):1287–1296.

Lall S, Marsden J and Glavaski S. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *Int. J. Robust Nonlinear Control* 2002; **12**:519–535.

Lamari H, Ammar A, Cartraud P, Legrain G, Jacquemin F and Chinesta F. Routes for efficient computational homogenization of non-linear materials using the proper generalized decomposition. *Arch. Comput. Methods Eng.* 2010; **17**(4):373–391.

Lamari H, Ammar A, Leygue A and Chinesta F. On the solution of the multidimensional Langer's equation by using the proper generalized decomposition method for modeling phase transitions. *Modell. Simul. Mater. Sci. Eng.* 2012; **20**(1):015007.

Lassila T, Manzoni A, Quarteroni A and Rozza G. Generalized reduced basis methods and n-width estimates for the approximation of the solution manifold of parametric PDEs. In *Analysis and Numerics of Partial Differential Equations*, Springer INdAM Series, Brezzi F, Colli Franzone P, Gianazza U and Gilardi G (eds). Springer-Verlag: Milan, 2013; 307–329.

Leblond C and Allery C. A priori space-time separated representation for the reduced order modeling of low Reynolds number flows. *Comput. Methods Appl. Mech. Eng.* 2014; **274**:264–288.

Le Bris C, Lelièvre T and Maday Y. Results and questions on a nonlinear approximation approach for solving high-dimensional partial differential equations. *Constr. Approx.* 2009; **30**:621–651.

Lee MYL. Estimation of the error in the reduced-basis method solution of differential algebraic equations. *SIAM J. Numer. Anal.* 1991; **28**:512–528.

Leonenko GM and Phillips TN. On the solution of the Fokker-Planck equation using a high-order reduced basis approximation. *Comput. Methods Appl. Mech. Eng.* 2009; **199**(1–4):158–168.

Leygue A, Chinesta F, Beringhier M, Nguyen TL, Grandidier JC, Pasavento F and Schrefler B. Towards a framework for non-linear thermal models in shell domains. *Int. J. Numer. Methods Heat Fluid Flow* 2013; **23**(1):55–73.

Leygue A and Verron E. A first step towards the use of proper general decomposition method for structural optimization. *Arch. Comput. Methods Eng.* 2010; **17**(4):I465–I472.

Lieu T, Farhat C and Lesoinne M. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Comput. Methods Appl. Mech. Eng.* 2006; **195**(41–43):5730–5742.

Loéve M. *Probability Theory*. D. Van Nostrand Company Inc.: New York, 1955.

Lubich Ch. *From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis*. European Mathematical Society: Zurich, 2008.

Lumley J. The structures of inhomogeneous turbulent flows. In *Atmospheric Turbulence and Radio Wave Propagation*, Yaglom AM and Tatarski VI (eds). NAUCA: Moscow1967; 166–178.

Maday Y, Patera AT and Turinici G. *A Priori* convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. *J. Sci. Comput.* 2002a; **17**(1-4):437–446.

Maday Y, Patera AT and Turinici G. Global *a priori* convergence theory for reduced-basis approximation of single-parameter symmetric coercive elliptic partial differential equations. *C.R. Acad. Sci. Paris, Sér. I* 2002b; **335**(3):289–294.

Manzoni A, Quarteroni A and Rozza G. Computational reduction for parametrized PDEs: strategies and applications. *Milan J. Math.* 2012; **80**:283–309.

Metoui S, Pruliere E, Ammar A, Dau F and Iordanoff I. The proper generalized decomposition for the simulation of delamination using cohesive zone model. *Int. J. Numer. Methods Eng.* 2014; **99**(13):1000–1022.

Modesto D, Zlotnik S and Huerta A. Proper generalized decomposition for parameterized Helmholtz problems in heterogeneous and unbounded domains: application to harbor agitation. *Comput. Methods Appl. Mech. Eng.* 2015; **295**:127–149.

Moitinho de Almeida JP. A basis for bounding the errors of proper generalised decomposition solutions in solid mechanics. *Int. J. Numer. Methods Eng.* 2013; **94**(10):961–984.

Mokdad B, Ammar A, Normandin M, Chinesta F and Clermont JR. A fully deterministic micro-macro simulation of complex flows involving reversible network fluid models. *Math. Comput. Simul.* 2010; **80**:1936–1961.

Mokdad B, Pruliere E, Ammar A and Chinesta F. On the simulation of kinetic theory models of complex fluids using the Fokker-Planck approach. *Appl. Rheol.* 2007; **17**(2), 26494, 1–14.

Nadal E, Leygue A, Chinesta F, Beringhier M, Rodenas JJ and Fuenmayor FJ. A separated representation of an error indicator for the mesh refinement process under the proper generalized decomposition framework. *Comput. Mech.* 2015; **55**(2):251–266.

Nazeer M, Bordeu F, Leygue A and Chinesta F. Arlequin based PGD domain decomposition. *Comput. Mech.* 2014; **54**(5):1175–1190.

Neron D, Boucard PA and Relun N. Time-space PGD for the rapid solution of 3D nonlinear parametrized problems in the many-query context. *Int. J. Numer. Methods Eng.* 2015; **103**(4):275–292.

Neron D and Dureisseix D. A computational strategy for poroelastic problems with a time interface between coupled physics. *Int. J. Numer. Methods Eng.* 2008a; **73**(6):783–804.

Neron D and Dureisseix D. A computational strategy for thermo-poroelastic structures with a time-space interface coupling. *Int. J. Numer. Methods Eng.* 2008b; **75**(9):1053–1084.

Néron D and Ladevèze P. Proper generalized decomposition for multiscale and multiphysics problems. *Arch. Comput. Methods Eng.* 2010; **17**(4):351–372.

Nguyen NC, Rozza G and Patera AT. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous burgers' equation. *Calcolo* 2009; **46**:157–185.

Nguyen NC, Veroy K and Patera AT. Certified real-time solution of parametrized partial differential equations. In *Handbook of Materials Modeling*, Yip S (ed.). Springer-Verlag: New York, 2005; 1523–1558.

Niroomandi S, Gonzalez D, Alfaro I, Bordeu F, Leygue A, Cueto E and Chinesta F. Real time simulation of biological soft tissues: a PGD approach. *Int. J. Numer. Methods Biomed. Eng.* 2013; **29**(5):586–600.

Noor AK. Recent advances in reduction methods for nonlinear problems. *Comput. Struct.* 1981; **13**:31–44.

Noor AK. On making large nonlinear problems small. *Comput. Meth. Appl. Mech. Eng.* 1982; **34**:955–985.

Noor AK, Balch CD and Shibut MA. Reduction methods for non-linear steady-state thermal analysis. *Int. J. Numer. Methods Eng.* 1984; **20**:1323–1348.

Noor AK and Peters JM. Reduced basis technique for nonlinear analysis of structures. *AIAA J.* 1980; **18**(4):455–462.

North G, Bell T, Cahalan R and Moeng F. Sampling errors in the estimation of empirical orthogonal functions. *Mon. Weather Rev.* 1982; **110**(7):699–706.

Nouy A. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Comput. Methods Appl. Mech. Eng.* 2007; **196**:4521–4537.

Nouy A. Generalized spectral decomposition method for solving stochastic finite element equations: invariant subspace problem and dedicated algorithms. *Comput. Methods Appl. Mech. Eng.* 2008; **197**:4718–4736.

Nouy A. Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Arch. Comput. Methods Eng.* 2009; **16**(3):251–285.

Nouy A. A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. *Comput. Methods Appl. Mech. Eng.* 2010a; **199**:1603–1626.

Nouy A. Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems. *Arch. Comput. Methods Eng. State Art Rev.* 2010b; **17**:403–434.

Nouy A. Fictitious domain method and separated representations for the solution of boundary value problems on uncertain parameterized domains. *Comput. Methods Appl. Mech. Eng.* 2011; **200**(45-46):3066–3082.

Nouy A and Falco A. Constrained Tensor Product Approximations based on penalized best approximations, 2011, oai:hal.archives-ouvertes.fr:hal-00577942.

Nouy A and Ladeveze P. Multiscale computational strategy with time and space homogenization: a radial type approximation technique for solving micro problems. *Int. J. Multiscale Comput. Eng.* 2004; **170**(2):557–574.

Nouy A and Le Maitre O. Generalized spectral decomposition method for stochastic non linear problems. *J. Comput. Phys.* 2009; **228**(1):202–235.

Oden JT, Belytschko T, Fish J, Hughes TJR, Johnson C, Keyes D, Laub A, Petzold L, Srolovitz D and Yip S. Simulation-based Engineering Science: Revolutionizing Engineering Science through simulation. NSF Blue Ribbon Panel on SBES, 2006.

Patera AT and Rozza G. *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations*, MIT Pappalardo Monographs in Mechanical Engineering, Copyright MIT (2006–2015), 2007, online at http://augustine.mit.edu.

Peherstorfer B, Butnaru D, Willcox K and Bungartz H. Localized discrete empirical interpolation method. *SIAM J. Sci. Comput.* 2014; **36**(1):A168–A192.

Peherstorfer B and Willcox K. Dynamic data-driven reduced-order models. *Comput. Methods Appl. Mech. Eng.* 2015a; **291**:21–41.

Peherstorfer B and Willcox K. Online adaptive model reduction for nonlinear systems via low-rank updates. *SIAM J. Sci. Comput.* 2015b; **37**(4):A2123–A2150.

Peterson JS. The reduced basis method for incompressible viscous flow calculations. *SIAM J. Sci. Stat. Comput.* 1989; **10**(4):777–786.

Porsching TA. Estimation of the error in the reduced basis method solution of nonlinear equations. *Math. Comput.* 1985; **45**(172):487–496.

Porsching TA and Lee MYL. The reduced-basis method for initial value problems. *SIAM J. Numer. Anal.* 1987; **24**:1277–1287.

Prud'homme C, Rovas D, Veroy K, Maday Y, Patera AT and Turinici G. Reliable real-time solution of parametrized partial differential equations: reduced-basis output bounds methods. *J. Fluids Eng.* 2002; **124**(1):70–80.

Pruliere E. 3D simulation of laminated shell structures using the proper generalized decomposition. *Compos. Struct.* 2014; **117**:373–381.

Pruliere E, Ammar A, El Kissi N and Chinesta F. Recirculating flows involving short fiber suspensions: numerical difficulties and efficient advanced micro-macro solvers. *Arch. Comput. Methods Eng. State Art Rev.* 2009; **16**:1–30.

Pruliere E, Chinesta F and Ammar A. On the deterministic solution of multidimensional parametric models by using the proper generalized decomposition. *Math. Comput. Simul.* 2010; **81**:791–810.

Quarteroni A and Rozza G. Numerical solution of parametrized Navier-Stokes equations by reduced basis method. *Numer. Meth. PDEs* 2007; **23**(4):923–948.

Quarteroni A, Rozza G and Manzoni A. Certified reduced basis approximation for parametrized PDE and applications. *J. Math Ind.* 2011; **1**(1):3.

Rathinam M and Petzold L. A new look at proper orthogonal decomposition. *SIAM J. Numer. Anal.* 2003; **41**(5):1893–1925.

Rheinboldt WC. Numerical analysis of continuation methods for nonlinear structural problems. *Comput. Struct.* 1981; **13**(1-3):103–113.

Rheinboldt WC. On the theory and error estimation of the reduced basis method for multi-parameter problems. *Nonlinear Anal. Theory Methods Appl.* 1993; **21**(11):849–858.

Riou H, Ladeveze P and Kovalevsky L. The variational theory of complex rays: an answer to the resolution of mid-frequency 3D engineering problems. *J. Sound Vib.* 2013; **332**:1947–1960.

Rowley C. Model reduction for fluids, using balanced proper orthogonal decomposition. *Int. J. Bifurcat. Chaos* 2005; **15**(3):997–1013.

Rozza G. Reduced-basis methods for elliptic equations in sub-domains with a posteriori error bounds and adaptivity. *Appl. Numer. Math.* 2005; **55**(4):403–424.

Rozza G. Reduced basis method for Stokes equations in domains with non-affine parametric dependence. *Comput. Vis. Sci.* 2009; **12**:23–25.

Rozza G. Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Commun. Comput. Phys.* 2011; **9**:1–48.

Rozza G. Fundamentals of reduced basis method for problems governed by parametrized PDEs and applications. In *CISM Lectures Notes "Separated Representation and PGD Based Model Reduction: Fundamentals and Applications"*, Ladeveze P and Chinesta F (eds). Springer-Verlag: Vienna, 2014.

Rozza G, Huynh P and Manzoni A. Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants. *Numer. Math.* 2013; **125**(1):115–152.

Rozza G, Huynh DBP and Patera AT. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Arch. Comput. Methods Eng.* 2008; **15**(3):229–275.

Rozza G and Veroy K. On the stability of reduced basis method for Stokes equations in parametrized domains. *Comput. Methods Appl. Mech. Eng.* 2007; **196**:1244–1260.

Schilders W, van der Vorst H and Rommes J (eds). *Model Order Reduction: Theory, Research Aspects and Applications*, *Mathematics in Industry*, vol. **13**. Springer-Verlag: Berlin, 2008.

Sirovich L. Turbulence and the dynamics of coherent structures. Part 1: Coherent structures. *Q. Appl. Math.* 1987; **45**(3):561–571.

Veroy K and Patera AT. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-basis *a posteriori* error bounds. *Int. J. Numer. Methods Fluids* 2005; **47**:773–788.

Veroy K, Prud'homme C and Patera AT. Reduced-basis approximation of the viscous Burgers equation: rigorous *a posteriori* error bounds. *C.R. Acad. Sci. Paris, Sér. I* 2003; **337**(9):619–624.

Vidal P, Gallimard L and Polit O. Composite beam finite element based on the proper generalized decomposition. *Comput. Struct.* 2012; **102**:76–86.

Vidal P, Gallimard L and Polit O. Proper generalized decomposition and layer-wise approach for the modeling of composite plate structures. *Int. J. Solids Struct.* 2013; **50**(14-15):2239–2250.

Vidal P, Gallimard L and Polit O. Explicit solutions for the modeling of laminated composite plates with arbitrary stacking sequences. *Composites Part B* 2014a; **60**:697–706.

Vidal P, Gallimard L and Polit O. Shell finite element based on the proper generalized decomposition for the modeling of cylindrical composite structures. *Comput. Struct.* 2014b; **132**:1–11.

Vidal P, Gallimard L and Polit O. Assessment of variable separation for finite element modeling of free edge effect for composite plates. *Compos. Struct.* 2015; **123**:19–29.

Vitse M, Neron D and Boucard PA. Virtual charts of solutions for parametrized nonlinear equations. *Comput. Mech.* 2014; **54**(6):1529–1539.

Volkwein S. *Model Reduction Using Proper Orthogonal Decomposition*, Lecture Notes. Institute of Mathematics and Scientific Computing, University of Graz, 2011, http://www.uni-graz.at/imawww/volkwein/POD.pdf.

Willcox K and Peraire J. Balanced model reduction via the proper orthogonal decomposition. *AIAA J.* 2002; **40**(11):2323–2330.

Willcox K, Peraire J and White J. An Arnoldi approach for generation of reduced-order models for turbomachinery. *Comput. Fluids* 2003; **31**(3):369–389.

Zlotnik S, Diez P, Modesto D and Huerta A. Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications. *Int. J. Numer. Methods Eng.* 2015; **103**(10):737–758.