NONINTRUSIVE APPROXIMATION OF PARAMETRIZED LIMITS OF MATRIX POWER ALGORITHMS – APPLICATION TO MATRIX INVERSES AND LOG-DETERMINANTS

Fabien Casenave[1,*], Nissrine Akkari[1], Alexandre Charles[1]
and Christian Rey[1]

Abstract. We consider in this work quantities that can be obtained as limits of powers of parametrized matrices, for instance the inverse matrix or the logarithm of the determinant. Under the assumption of affine dependence in the parameters, we use the Empirical Interpolation Method (EIM) to derive an approximation for powers of these matrices, from which we derive a nonintrusive approximation for the aforementioned limits. We derive upper bounds of the error made by the obtained formula. Finally, numerical comparisons with classical intrusive and nonintrusive approximation techniques are provided: in the considered test-cases, our algorithm performs well compared to the nonintrusive ones.

1. Introduction

Many models in physics, biology or engineering involve partial differential equations, which are nowadays mainly solved numerically. In many cases, a single solution is not enough, as we are interested in the behavior of the solution when some chosen parameters vary. For instance, in sensitivity analyses, optimization or uncertainty quantification, the solution has to be computed a large number of times. In this many-queries context, model order reduction techniques have proved to allow large improvements in computational costs.

A number of techniques and methods can be grouped under the heading of Reduced Order Models (ROM). First, one can simply consider taking a coarser mesh, or making use of symmetries in the problem. Then, one can use methods from the machine learning community, where a meta-model is constructed as an interpolation or regression of the solutions or quantities of interest over the parameter set. These techniques are nonintrusive since they use the numerical solver as a black-box, see [1,16,18,24,31] for reviews of machine learning regression methods. Finally, a third class of ROM consists in solving the partial differential equation (or an approximation of it) on a small dimensional subspace, so that the computational cost of solving the reduced model is orders of magnitude smaller than that of the full-scale model. For instance, one can use the Proper Orthogonal Decomposition (POD) [10,30] or the Reduced-Basis method [6,17,19,20,25,28,29,32,33,35]. The Proper Generalized Decomposition (PGD) usually expresses the solution as a function of space and time, and the parameters of

[1] Safran Tech, Rue des Jeunes Bois, Châteaufort CS 80112 – 78772 Magny-les-Hameaux Cedex, France.
*Corresponding author: fabien.casenave@safrangroup.com

the model, seen here as coordinates. This function is approximated as a sum of tensor products, see [12–15]. These methods are generally very intrusive to the considered computational code, since they need to modify the assembly routines of the operators. Efforts have been spent to mitigate these intrusivity requirements [7, 9, 34], but we still need to manipulate at least the matrices or meshes.

In this work, we consider a family of parametrized invertible matrices, such that the parameter dependence is affine, and we are interested in the nonintrusive approximation of quantities obtained as limits of powers of parametrized matrices, for instance the inverse matrices or the logarithm of the determinant (log-det), which we express as linear combinations of these inverses or log-det computed at given parameter values. For instance, many evaluations of the log-det of a positive-definite matrix are required for maximum likelihood estimation in Gaussian process regression, see [23, 37]. The proposed algorithm computes the coefficients of these linear combinations efficiently (namely in a computational complexity independent of the size of the matrices), and is nonintrusive, in the sense that it resorts only to the evaluation of the quantities of interests, namely the inverse or the log-det, like most machine learning methods. This is an offline/online procedure. A computationally demanding stage is first carried out, the offline stage, where the high-fidelity model is solved a certain number of times and some information on the parameter dependence of the model is learned. This information is then exploited in the online stage, in a computationally cheap fashion, where the approximation is computed rapidly and potentially for a large number of parameter values.

In Section 2 is proposed an interpolation formula of the power of parametrized matrices based on the Empirical Interpolation Method (EIM) [3, 22]. Then in Section 3, nonintrusive approximations of limits to certain power algorithms are obtained using the interpolation of power of matrices, namely the inverse and the log-det of parametrized matrices. In Section 4, upper bounds of the error made by these approximations are derived. Finally, in Section 5 are presented numerical comparisons between the proposed nonintrusive approximation algorithm and classical intrusive and nonintrusive methods.

2. Approximation of powers of parametrized matrices

Let $d \in \mathbb{N}$ and $\mu \in \mathcal{P}$ be a parameter, where the parameter set $\mathcal{P}$ is a compact subset of $\mathbb{R}^r$, $r \in \mathbb{N}^*$. Consider $\{A_\mu\}_{\mu \in \mathcal{P}} \subset \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ a set of parametrized square matrices and assume the following affine decomposition for each element of the set:

$$A_\mu = \sum_{l=1}^{d} \alpha_l(\mu) A_l, \ \mu \in \mathcal{P}, \tag{2.1}$$

where we suppose that the family of square matrices $\{A_l\}_{1 \le l \le d}$ is independent of μ. Hence, the matrix A_μ depends on μ only through the coefficients $\alpha_l : \mathcal{P} \to \mathbb{R}$.

Let $m \in \mathbb{N}$. We propose to derive an offline/online procedure to compute an approximation of A_μ^p for $1 \le p \le m$ and $\mu \in \mathcal{P}$ of the following form:

$$A_\mu^p \approx \sum_{l=1}^{t} \lambda_l(\mu) A_{\mu_l}^p, \ \mu \in \mathcal{P}, \tag{2.2}$$

where $\{\mu_l\}_{1 \le l \le t}$ is determined during the offline stage whereas the applications $\lambda_l : \mathcal{P} \to \mathbb{R}$ are computed during the online stage. As we shall see later, this expression will be used to obtain efficient approximations for the inverse and the log-det of A_μ.

Consider the decomposition (2.1) and take the pth power of the equation. In the general case, the matrices A_i, $1 \le i \le d$ do not commute, which prevents us from the use of the multinomial theorem. Thus,

$$A_\mu^p = \sum_{l_1=1}^{d} \sum_{l_2=1}^{d} \cdots \sum_{l_p=1}^{d} \left(\prod_{i=1}^{p} \alpha_{l_i}(\mu) \right) \left(\prod_{i=1}^{p} A_{l_i} \right), \ \mu \in \mathcal{P}. \tag{2.3}$$

In the following, we factorize the sum according to the products of A_l matrices. Let $t \in \mathbb{N}$. Denote a multi-index $\vec{s} = (s_1, s_2, \cdots, s_t) \in \mathbb{N}^t$ and define its weight $|\vec{s}| := \sum_{l=1}^{t} s_l$. Finally, denote $\overline{\kappa}_{m,d} = \left\{ \vec{k} \in [\![0; m]\!]^d \text{ such that } |\vec{k}| \leq m \right\}$.

Lemma 2.1. *Let p, d, $m \in \mathbb{N}$ and $\{A_\mu\}_{\mu \in \mathcal{P}} \subset \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ a set of parametrized matrices satisfying* (2.1). *There exists $\{T_{\vec{k},p}\}_{\vec{k} \in \overline{\kappa}_{m,d}, 0 \leq p \leq m} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$, independent of μ, such that the following equality holds:*

$$A_\mu^p = \sum_{\vec{k} \in \overline{\kappa}_{m,d}} g(\vec{k}, \mu) T_{\vec{k},p}, \ \mu \in \mathcal{P}, \ 1 \leq p \leq m, \tag{2.4}$$

where

$$g(\vec{k}, \mu) = \prod_{l=1}^{d} \alpha_l^{k_l}(\mu). \tag{2.5}$$

We clarify here that Lemma 2.1 contains an existence result for the matrices $T_{\vec{k},p}$, that do not need to be computed for the method to be carried out in practice. Equation (2.4) indicates that the function $g \mapsto A^p$ is linear, and the idea is to use an EIM approximation of $g(\vec{k}, \mu)$ to readily obtain an approximation of A_μ^p. Working on $g(\vec{k}, \mu)$ instead of the matrix coefficients $\left(A_\mu^p\right)_{i,j}$, $1 \leq i, j \leq \mathcal{N}$, will enable us to construct nontrusive approximations for the inverse or log-det of A_μ^p, as we will see in Section 3.

Proof of Lemma 2.1. Define the family of applications:

$$\mathcal{F}_{p,d} : \quad \begin{cases} [\![1; d]\!]^p \to [\![0; p]\!]^d \\ \vec{s} \mapsto (\# \{s_i = j, \ 1 \leq i \leq p\})_{j=1,\cdots,d} \end{cases} \quad \forall d, q \in \mathbb{N}. \tag{2.6}$$

By construction, $|\mathcal{F}_{p,d}(\vec{s})| = p$ for all $\vec{s} \in [\![1; d]\!]^p$. Take $d = 2$, $p = 3$ for example. Then $\mathcal{F}_{3,2}((2,2,2)) = (0,3)$ and $\mathcal{F}_{3,2}((1,2,2)) = \mathcal{F}_{3,2}((2,1,2)) = \mathcal{F}_{3,2}((2,2,1)) = (1,2)$.

Define now the reciprocal applications:

$$\mathcal{I}_{p,d} : \quad \begin{cases} [\![0; p]\!]^d \to \wp \left([\![1; d]\!]^p\right) \\ \vec{k} \mapsto \left\{ \vec{s} \in [\![1; d]\!]^p \ : \ \mathcal{F}_{p,d}(\vec{s}) = \vec{k} \right\} \end{cases} \quad \forall d, q \in \mathbb{N} \tag{2.7}$$

where $\wp \left([\![1; d]\!]^p\right)$ denotes the power set of $[\![1; d]\!]^p$. Take $d = 2$, $p = 3$ for example. Then, $\mathcal{I}_{3,2}((0,3)) = \{(2,2,2)\}$ or $\mathcal{I}_{3,2}((1,2)) = \{(1,2,2), (2,1,2), (2,2,1)\}$.

Using the introduced notation, equation (2.3) can be reordered in the following form:

$$A_\mu^p = \sum_{\vec{k} \in [\![0; p]\!]^d : |\vec{k}| = p} \left(\prod_{l=1}^{d} \alpha_l^{k_l}(\mu) \right) \sum_{\vec{s} \in \mathcal{I}_{p,d}(\vec{k})} \prod_{i=1}^{p} A_{s_i}, \quad \forall \mu \in \mathcal{P}. \tag{2.8}$$

Notice that if the matrices A_l, $1 \leq l \leq d$, were commuting, we could have simply applied the multinomial theorem to get

$$A_\mu^p = \sum_{\vec{k} \in [\![0; p]\!]^d : |\vec{k}| = p} \frac{p!}{\prod_{l=1}^{d} k_l!} \left(\prod_{l=1}^{d} \alpha_l^{k_l}(\mu) \right) \left(\prod_{l'=1}^{d} A_{l'}^{k_{l'}}(\mu) \right), \quad \forall \mu \in \mathcal{P}. \tag{2.9}$$

Recall the notation $g(\vec{k}, \mu) = \prod_{l=1}^{d} \alpha_l^{k_l}(\mu)$ and denote

$$\tilde{T}_{\vec{k},p} := \sum_{\vec{s} \in \mathcal{I}_{p,d}(\vec{k})} \prod_{i=1}^{p} A_{s_i}, \tag{2.10}$$

so that

$$A_\mu^p = \sum_{\vec{k} \in [\![0;p]\!]^d : |\vec{k}| = p} g(\vec{k}, \mu) \tilde{T}_{\vec{k},p}, \; \mu \in \mathcal{P}. \tag{2.11}$$

Denote now for a general $\vec{k} \in [\![0;p]\!]^d$ (not restricted to only $|\vec{k}| = p$):

$$T_{\vec{k},p} := \begin{cases} \tilde{T}_{\vec{k},p} & \text{if } |\vec{k}| = p, \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

Let $m \in \mathbb{N}$. The p-exponent in (2.11) can be parametrized using

$$A_\mu^p = \sum_{\vec{k} \in \overline{\kappa}_{m,d}} g(\vec{k}, \mu) T_{\vec{k},p}, \; \mu \in \mathcal{P}, \quad 1 \le p \le m, \tag{2.13}$$

where we recall that $\overline{\kappa}_{m,d} = \left\{ \vec{k} \in [\![0;m]\!]^d \text{ such that } |\vec{k}| \le m \right\}$, which concludes the proof. $\qquad\square$

To illustrate Lemma 2.1, consider the case $p = 2$ and $d = 2$. In this case, $\left\{ \vec{k} \in [\![0;2]\!]^2 : |\vec{k}| = 2 \right\} = \{(1,1), (0,2), (2,0)\}$, and from equations (2.12) and (2.13), there holds

$$A_\mu^2 = g((1,1), \mu) T_{(1,1),2} + g((0,2), \mu) T_{(0,2),2} + g((2,0), \mu) T_{(2,0),2}. \tag{2.14}$$

Since $\mathcal{F}_{2,2}((1,2)) = \mathcal{F}_{2,2}((2,1)) = (1,1)$, there holds $\mathcal{I}_{2,2}((1,1)) = \{(1,2), (2,1)\}$, and from equation (2.10), $\tilde{T}_{(1,1),2} = A_1 A_2 + A_2 A_1$. In the same fashion, we compute $\mathcal{F}_{2,2}((2,2)) = (0,2)$ leading to $\mathcal{I}_{2,2}((0,2)) = \{(2,2)\}$ and $\tilde{T}_{(0,2),2} = A_2^2$, as well as $\mathcal{F}_{2,2}((1,1)) = (2,0)$ leading to $\mathcal{I}_{2,2}((2,0)) = \{(1,1)\}$ and $\tilde{T}_{(2,0),2} = A_1^2$. Using the formula (2.5) for g in (2.14) leads to the known expression $A_\mu^2 = \alpha_1(\mu) \alpha_2(\mu) (A_1 A_2 + A_2 A_1) + \alpha_1(\mu)^2 A_1^2 + \alpha_2(\mu)^2 A_2^2$.

It is known that $\#\{\vec{k} \in [\![0;p]\!]^d \text{ such that } |\vec{k}| = p\} = \frac{(p+d-1)!}{(d-1)!p!}$. Then, the number of terms in (2.4), namely $Q_{m,d} := \#\overline{\kappa}_{m,d}$, equals $\frac{1}{(d-1)!} \sum_{k=0}^{m} \frac{(k+d-1)!}{k!}$. Notice that for $d \ge 2$, $Q_{m,d} \le \frac{m}{(d-1)!} (m+1)(m+2) \cdots (m+d-1) = P_d(m)$, where $P_d(m)$ is a polynomial of degree d in m.

As discussed earlier, we carry out the Empirical Interpolation Method (EIM) on the $g(\vec{k}, \mu)$, see Algorithm 1 for a description of the offline stage of EIM on this function. In Algorithm 1, $\delta^l g := I^l(g) - g$, with $I^l(g)$ denoting the rank-l EIM approximation, defined by

$$I^l(g)(\vec{k}, \mu) = \sum_{l'=1}^{l} \beta_{l'}^l(\mu) q^{l'}(\vec{k}), \tag{2.15}$$

where $\beta^l(\mu)$ solves

$$\sum_{l''=1}^{l} B_{l',l''}^l \beta_{l''}^l(\mu) = g(\vec{k}_{l'}, \mu), \qquad 1 \le l' \le l. \tag{2.16}$$

The quantities $B^l \in \mathbb{R}^{l \times l}$, $q^l : \mathcal{P} \to \mathbb{R}^l$, $\vec{k}_l \in \overline{\kappa}_{m,d}$, $\mu_l \in \mathcal{P}_{\text{sample}}$, for all $1 \le l \le N^{\text{EIM}}$, are constructed during the offline stage in Algorithm 1, where N^{EIM} is the number of terms selected by the EIM. In practice, N^{EIM} is not *a priori* specified, but results from a stopping criterion on the maximum current error $(\delta^l g)(\vec{k}_{l+1}, \mu_{l+1})$ made by the approximation.

Finally, the online stage of EIM consists in the approximation (2.15) and (2.16) with $l = N^{\text{EIM}}$. Replacing $\beta_{l'}^l(\mu)$ in equation (2.15) using equation (2.16) yields

$$I^{N^{\text{EIM}}}(g)(\vec{k}, \mu) = \sum_{l'=1}^{N^{\text{EIM}}} \sum_{l''=1}^{N^{\text{EIM}}} \left(B^{N^{\text{EIM}}} \right)_{l',l''}^{-1} g(\vec{k}_{l''}, \mu) q^{l'}(\vec{k}). \tag{2.17}$$

Algorithm 1 Offline stage of the EIM

---

1. Choose a fine finite set $\mathcal{P}_{\text{sample}} \subset \mathcal{P}$
2. Set $l := 1$
3. Compute $\mu_1 := \underset{\mu \in \mathcal{P}_{\text{sample}}}{\text{argmax}} \|g(\cdot, \mu)\|_{\ell^\infty(\overline{\kappa}_{m,d})}$
4. Compute $\vec{k}_1 := \underset{\vec{k} \in \overline{\kappa}_{m,d}}{\text{argmax}} |g(\vec{k}, \mu_1)|$
5. Set $q^1(\cdot) := \dfrac{g(\cdot, \mu_1)}{g(\vec{k}_1, \mu_1)}$
6. Set $B_{11}^1 := 1$
7. **while** $l < Q_{m,d}$ **do**
8.   Compute $\mu_{l+1} := \underset{\mu \in \mathcal{P}_{\text{sample}}}{\text{argmax}} \|(\delta^l g)(\cdot, \mu)\|_{\ell^\infty(\overline{\kappa}_{m,d})}$
9.   Compute $\vec{k}_{l+1} := \underset{\vec{k} \in \overline{\kappa}_{m,d}}{\text{argmax}} |(\delta^l g)(\vec{k}, \mu_{l+1})|$
10.   Set $q^{l+1}(\cdot) := \dfrac{(\delta^l g)(\cdot, \mu_{l+1})}{(\delta^l g)(\vec{k}_{l+1}, \mu_{l+1})}$
11. $B_{ij}^{l+1} := q_j(\vec{k}_i), \ 1 \le i, j \le l+1$
12. $l \leftarrow l + 1$
13. **end while**

We notice from Algorithm 1 that $\underset{1 \le l \le N^{\text{EIM}}}{\text{Span}} \left(q^l(\cdot) \right) = \underset{1 \le l \le N^{\text{EIM}}}{\text{Span}} \left(g(\cdot, \mu_l) \right)$, and therefore, there exists a matrix $\Gamma \in \mathbb{R}^{N^{\text{EIM}} \times N^{\text{EIM}}}$ such that, for all $1 \le l \le N^{\text{EIM}}$,

$$\sum_{l'=1}^{N^{\text{EIM}}} \Gamma_{l,l'} q^{l'}(\vec{k}) = g(\vec{k}, \mu_l). \tag{2.18}$$

Replacing $q^{l'}(\vec{k})$ in equation (2.17) using equation (2.18) yields

$$I^{N^{\text{EIM}}}(g)(\vec{k}, \mu) = \sum_{l'=1}^{N^{\text{EIM}}} \sum_{l=1}^{N^{\text{EIM}}} \Delta_{l,l'} g(\vec{k}_{l'}, \mu) g(\vec{k}, \mu_l), \tag{2.19}$$

where $\Delta = \left(\Gamma \left(B^{N^{\text{EIM}}} \right)^t \right)^{-1}$. From Theorem 1.2 of [8], the matrix $F_{l,l'} = g(\vec{k}_l, \mu_{l'}), \ 1 \le l, l' \le N^{\text{EIM}}$ is invertible, and $\Delta = F^{-T}$. Now denote

$$\lambda_l^{N^{\text{EIM}}}(\mu) = \sum_{l'=1}^{N^{\text{EIM}}} \Delta_{l,l'} g(\vec{k}_{l'}, \mu) \tag{2.20}$$

to obtain

$$I^{N^{\text{EIM}}}(g)(\vec{k}, \mu) = \sum_{l=1}^{N^{\text{EIM}}} \lambda_l^{N^{\text{EIM}}}(\mu) g(\vec{k}, \mu_l). \tag{2.21}$$

Replacing g in (2.4) by $I^{N^{\mathrm{EIM}}}(g)$ yields

$$A_\mu^p \approx \sum_{\vec{k}\in\overline{\kappa}_{m,d}} \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l^{N^{\mathrm{EIM}}}(\mu) g(\vec{k},\mu_l) T_{\vec{k},p} = \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l^{N^{\mathrm{EIM}}}(\mu) \sum_{\vec{k}\in\overline{\kappa}_{m,d}} g(\vec{k},\mu_l) T_{\vec{k},p}$$

$$= \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l^{N^{\mathrm{EIM}}}(\mu) A_{\mu_l}^p, \quad \mu \in \mathcal{P}, \ 1 \le p \le m, \tag{2.22}$$

where the last equality is obtained by recognizing $A_{\mu_l}^p$ in equation (2.4) at parameter values μ_l.

We obtain the searched expression (2.2), with $t = N^{\mathrm{EIM}}$, and where μ_l and $\lambda_l(\mu)$ are constructed in respectively the offline and online stages of an EIM on the $g(\vec{k},\mu)$:

$$A_\mu^p \approx \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l^{N^{\mathrm{EIM}}}(\mu) A_{\mu_l}^p, \quad \mu \in \mathcal{P}, \ 1 \le p \le m. \tag{2.23}$$

Notice also that the offline stage of EIM involves a sampling of $\mathcal{P}$, and $Q_{m,d}$ indices. Even though $Q_{m,d}$ is independent of the size N of the matrix A_μ, we are limited to moderate values of m and d in practice.

The expression (2.23) can be used to readily approximate any quantity expressed as a linear evaluation of power of matrices. Yet, quantities such as the inverse matrix or the logarithm of the determinant can be approximated using algorithms involving successive powers of the considered matrix. Hence, we combine in the next section these techniques with the approximation presented in the present section. Under particular conditions, inverses and logarithm of determinants of matrices can be approximated in a nonintrusive fashion using (2.23).

3. Power algorithms

3.1. Inverse operators and solution to linear systems

We recall hereby a classical fixed point results.

Lemma 3.1. *Let A and $\Psi \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$. We consider the following iterative scheme:*

$$\begin{cases} X_0 = X^0 \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}} \\ X_{k+1} = \left(I - \Psi^{-1}A\right)X_k + \Psi^{-1}. \end{cases} \tag{3.1}$$

If Ψ is chosen such that $\|I - \Psi^{-1}A\|_2 < 1$, then the sequence X_k converges towards A^{-1} for any initial guess X^0.

3.1.1. Sequence approximating the inverse of parametrized matrices

For concrete implementation, the kth iteration can be evaluated as a series of powers of A and provides an approximation of A^{-1}. For a parameter indexed family of matrices, we combine this approximation technique with results of the previous section.

Consider now a parameter-dependent family of matrices A_μ, $\mu \in \mathcal{P}$, verifying the affine decomposition (2.1), and such that, for all $\mu \in \mathcal{P}$, A_μ is invertible. We construct a family of approximations of the inverses, $(X_{k,\mu})_{k\in\mathbb{N}}$. To obtain a uniform convergence with respect to μ, we are led to choose a preconditioner uniform in μ and a common initial condition: denote

$$\begin{cases} \Psi_0 = \underset{M\in\mathbb{R}^{\mathcal{N}\times\mathcal{N}}, \ M\text{invertible}}{\operatorname{argmin}} \ \underset{\mu\in\mathcal{P}}{\sup} \ \|I - M^{-1}A_\mu\|_2, \\ X_0 = \underset{X\in\mathbb{R}^{\mathcal{N}\times\mathcal{N}}}{\operatorname{argmin}} \ \underset{\mu\in\mathcal{P}}{\sup} \ \|X - A_\mu^{-1}\|_2, \end{cases} \tag{3.2}$$

and let

$$\rho = \sup_{\mu \in \mathcal{P}} \|I - \Psi_0^{-1} A_\mu\|_2, \tag{3.3}$$

and $\epsilon_0 = \sup_{\mu \in \mathcal{P}} \|X_0 - A_\mu^{-1}\|_2$.

We suppose that $\rho < 1$ and consider the following iterative scheme:

$$\begin{cases} X_{0,\mu} = X_0 \\ X_{k+1,\mu} = \left(I - \Psi_0^{-1} A_\mu\right) X_{k,\mu} + \Psi_0^{-1}. \end{cases} \tag{3.4}$$

An induction shows that:

$$X_{m,\mu} = \left(I - \Psi_0^{-1} A_\mu\right)^m \left(X_0 - A_\mu^{-1}\right) + A_\mu^{-1}. \tag{3.5}$$

Taking the norm, we get a uniform bound with respect to μ:

$$\|X_{m,\mu} - A_\mu^{-1}\|_2 \leq \epsilon_0 \rho^m, \tag{3.6}$$

which ensures convergence with respect to m since $\rho < 1$.

3.1.2. Powers of a parametrized matrix

Define $\alpha_0(\mu) = 1$ and $A_0 = -\Psi_0$. There holds:

$$\left(I - \Psi_0^{-1} A_\mu\right) = -\sum_{l=0}^{d} \alpha_l(\mu) \Psi_0^{-1} A_l. \tag{3.7}$$

Apply Lemma 2.1 to $\left(I - \Psi_0^{-1} A_\mu\right)$ to get

$$\left(I - \Psi_0^{-1} A_\mu\right)^p = \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} \hat{g}\left(\vec{k}, \mu\right) \hat{T}_{\vec{k},p}, \tag{3.8}$$

where $\hat{g}(\vec{k}, \mu) = \prod_{l=0}^{d} \alpha_l^{k_l}(\mu)$, and $\hat{T}_{\vec{k},p}$ are independent of μ (the $-$ signs being integrated to the $\hat{T}_{\vec{k},p}$). There holds $\forall \mu \in \mathcal{P}$, $\forall \vec{k} \in \overline{\kappa}_{m,d+1}$, $\hat{g}(\vec{k}, \mu) = g(c(\vec{k}), \mu)$, where c cuts the 0th element of $\vec{k} \in \overline{\kappa}_{m,d+1}$. Notice that $c(\vec{k})$ belongs to $\overline{\kappa}_{m,d}$. Hence, we can apply the EIM approximation to the function $g(c(\vec{k}), \mu)$ on $\overline{\kappa}_{m,d} \times \mathcal{P}$ (see (2.21)) to obtain:

$$\left(I - \Psi_0^{-1} A_\mu\right)^p \approx \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} \left(\sum_{l=1}^{N^{\text{EIM}}} \lambda_l(\mu) g\left(c(\vec{k}), \mu_l\right)\right) \hat{T}_{\vec{k},p}. \tag{3.9}$$

We now switch the summations to obtain the desired result:

$$\left(I - \Psi_0^{-1} A_\mu\right)^p \approx \sum_{l=1}^{N^{\text{EIM}}} \lambda_l(\mu) \left(I - \Psi_0^{-1} A_{\mu_l}\right)^p, \quad \mu \in \mathcal{P}, \quad 1 \leq p \leq m, \tag{3.10}$$

where we recall that μ_l and $\lambda_l(\mu)$ are given by EIM on $g(\vec{k}, \mu)$.

Notice that (3.10) means that we used the same EIM on $g(\vec{k}, \mu)$ for the affine approximation (2.1) on A_μ to get an approximation on $\left(I - \Psi_0^{-1} A_\mu\right)^p$.

3.1.3. *Approximation of the inverse of parametrized matrices*

We now go back to the scheme (3.4) and to show how the approximations (3.10) can be used to approximate the mth approximation $X_{m,\mu}$. Expression (3.5) is not convenient for this purpose since A_μ^{-1} appears, and we are looking for an efficient approximation of A_μ^{-1}. It turns out to be more convenient to consider the following expression obtained by induction:

$$X_{m,\mu} = \left(I - \Psi_0^{-1} A_\mu\right)^m X_0 + \left(\sum_{k=0}^{m-1} \left(I - \Psi_0^{-1} A_\mu\right)^k\right) \Psi_0^{-1}, \quad \forall m \in \mathbb{N}. \tag{3.11}$$

To obtain (3.10), we use the fact that $\left(I - \Psi_0^{-1} A_\mu\right)^p$ depends linearly on g, as it explicitly appears in (3.8). $X_{m,\mu}$ inherits from this linear dependence on g; we will make it explicit be denoting now $X_{m,\mu}$ as $X_m g_\mu$, where $g_\mu(\vec{k}) := g(c(\vec{k}), \mu)$.

Now replace the powers of $\left(I - \Psi_0^{-1} A_\mu\right)$ in (3.11) using (3.10):

$$\begin{aligned}
X_{m,\mu} = X_m g_\mu &\approx \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) \left(I - \Psi_0^{-1} A_{\mu_l}\right)^m X_0 + \sum_{k=0}^{m-1} \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) \left(I - \Psi_0^{-1} A_{\mu_l}\right)^k \Psi_0^{-1} \\
&= \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) \left(\left(I - \Psi_0^{-1} A_{\mu_l}\right)^m X_0 + \sum_{k=0}^{m-1} \left(I - \Psi_0^{-1} A_{\mu_l}\right)^k \Psi_0^{-1}\right) \\
&= \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) X_m g_{\mu_l}.
\end{aligned} \tag{3.12}$$

The convergence of $X_m g_\mu$ to A_μ^{-1} with respect to m, namely (3.6), suggests replacing $X_m g_{\mu_l}$ by the inverses $A_{\mu_l}^{-1}$ in (3.12) and defining

$$\mathcal{X}_\mu^{N^{\mathrm{EIM}}} := \sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) A_{\mu_l}^{-1}. \tag{3.13}$$

where $\mathcal{X}_\mu^{N^{\mathrm{EIM}}}$ is the obtained approximation of A_μ^{-1}.

Remark 3.2 (Nonintrusivity)**.** In equation (3.13), we recall that the coefficients $\lambda_l(\mu)$ are obtained from an EIM on $g(c(\vec{k}), \mu)$, which only depends on the parametric dependance of A_{μ_l}, see equations (2.1) and (2.5). Therefore, the obtained approximation is nonintrusive in the sense that we only resort to the computation of the quantity of interest (the inverses $A_{\mu_l}^{-1}$) and using some knowledge on the particular form of the problem (the $\alpha_l(\mu)$). In particular, we need to compute neither the $\left(I - \Psi_0^{-1} A_\mu\right)^p$ nor the $\hat{T}_{\vec{k},p}$. Notice that the matrices A_l in (2.1), $1 \le l \le d$, do not need to be computed either. Moreover, we can compute $A_{\mu_l}^{-1}$ by the method of our choice. Even if the described iterative scheme converges, we can use direct methods to compute the $A_{\mu_l}^{-1}$, and apply (3.13) to retrieve an approximation of A_μ^{-1}. We can also compute $A_{\mu_l}^{-1}$ using initial μ-dependent initial guesses $X_{0,\mu}$ and preconditioner $\Psi_{0,\mu}$ in the described iterative scheme. In particular, we never need to construct Ψ_0 and X_0, we just need the existence of a matrix $\hat{\Psi}$ such that $\sup_{\mu \in \mathcal{P}} \|I - \hat{\Psi} A_\mu\|_2 < 1$.

Remark 3.3 (Solution of linear systems)**.** Let $b \in \mathbb{R}^{\mathcal{N}}$. From (3.6), there holds $\|X_{k,\mu} b - A_\mu^{-1} b\|_2 \le \rho^k \epsilon_0 \|b\|_2$, which suggests to approximating $A_\mu^{-1} b$ by

$$\sum_{l=1}^{N^{\mathrm{EIM}}} \lambda_l(\mu) \left(A_{\mu_l}^{-1} b\right). \tag{3.14}$$

Notice that a key element of the section is the linearity of the function $g \mapsto X_m g_\mu$, where, from (3.8) and (3.11),

$$X_m g_\mu = \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} g(c(\vec{k}), \mu) \left(\hat{T}_{\vec{k},m} X_0 + \sum_{l=0}^{m-1} \hat{T}_{\vec{k},l} \Psi_0^{-1} \right). \tag{3.15}$$

3.2. Logarithm of the determinant

The logarithm of the determinant (log-det) of a symetric positive definite (SPD) matrix is a quantity receiving interest in the literature. For instance, finding the maximum likelihood estimator of the mean and the covariance matrix of a normal multivariate distribution involves the computation of the log-det of a SPD matrix, see equation (2.2) form [2].

3.2.1. Sequence approximating the logarithm of the determinant of parametrized matrices

Consider a family of parametrized SPD matrices $A_\mu \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$ and denote $\rho(A_\mu)$ the spectral radius of A_μ. Suppose that $\sup_{\mu \in \mathcal{P}} \rho(A_\mu) < \infty$ and that we can determine some $\rho_M > \sup_{\mu \in \mathcal{P}} \max_{1 \leq i \leq \mathcal{N}} r_i(A_\mu)$, where $\{r_i(A_\mu)\}_{1 \leq i \leq \mathcal{N}}$ denotes the set of eigenvalues of A_μ. Denote $\rho_0 := \inf_{\mu \in \mathcal{P}} \min_{1 \leq i \leq \mathcal{N}} r_i(A_\mu)$, that we suppose strictly positive. From Lemma 5 of [5],

$$\log(\det(A_\mu)) = \mathcal{N} \log(\rho_M) - \sum_{k=1}^{\infty} \frac{\operatorname{tr}\left((I - \frac{1}{\rho_M} A_\mu)^k\right)}{k}. \tag{3.16}$$

Let $m \in \mathbb{N}$ and consider the following approximation of $\log(\det(A_\mu)) - \mathcal{N} \log(\rho_M)$:

$$X_m g_\mu := -\sum_{k=1}^{m-1} \frac{\operatorname{tr}\left((I - \frac{1}{\rho_M} A_\mu)^k\right)}{k}, \tag{3.17}$$

where we already make explicit the linear dependence in g (see (3.8)).

Since A_μ is SPD, there exists a family of unitary matrices U_μ such that $A_\mu = U_\mu D_\mu U_\mu^T$, D_μ being a diagonal matrix such that $D_{\mu_{i,i}} = r_i(A_\mu)$. From $\left(I - \frac{1}{\rho_M} A_\mu\right) = U_\mu \left(I - \frac{1}{\rho_M} D_\mu\right) U_\mu^T$, there holds $\operatorname{tr}\left((I - \frac{1}{\rho_M} A_\mu)^k\right) = \sum_{i=1}^{\mathcal{N}} \left(1 - \frac{r_i(A_\mu)}{\rho_M}\right)^k$, from which we infer

$$\begin{aligned}
|X_m g_\mu - (\log(\det(A_\mu)) - \mathcal{N} \log(\rho_M))| &= \sum_{k=m}^{\infty} \frac{\operatorname{tr}\left((I - \frac{1}{\rho_M} A_\mu)^k\right)}{k} \\
&\leq \mathcal{N} \sum_{k=m}^{\infty} \frac{\left(1 - \frac{\rho_0}{\rho_M}\right)^k}{k} \\
&\leq \frac{\mathcal{N}}{m} \sum_{k=m}^{\infty} \left(1 - \frac{\rho_0}{\rho_M}\right)^k.
\end{aligned} \tag{3.18}$$

Notice that

$$\begin{aligned}
\frac{\rho_0}{\rho_M} \sum_{k=m}^{\infty} \left(1 - \frac{\rho_0}{\rho_M}\right)^k &= \left[1 - \left(1 - \frac{\rho_0}{\rho_M}\right)\right] \sum_{k=m}^{\infty} \left(1 - \frac{\rho_0}{\rho_M}\right)^k \\
&= \sum_{k=m}^{\infty} \left(1 - \frac{\rho_0}{\rho_M}\right)^k - \sum_{k=m+1}^{\infty} \left(1 - \frac{\rho_0}{\rho_M}\right)^k \\
&= \left(1 - \frac{\rho_0}{\rho_M}\right)^m.
\end{aligned} \tag{3.19}$$

Injecting (3.19) in the last term of (3.18), we obtain

$$|X_m g_\mu + \mathcal{N}\log(\rho_M) - \log(\det(A_\mu))| \leq \mathcal{N}\frac{\rho_M}{\rho_0}\frac{\left(1 - \frac{\rho_0}{\rho_M}\right)^m}{m}, \tag{3.20}$$

which ensures convergence with respect to m since $0 \leq 1 - \frac{\rho_0}{\rho_M} < 1$.

3.2.2. Powers of a parametrized matrix

Define $\alpha_0(\mu) = 1$ and $A_0 = -aI$. There holds:

$$\left(I - \frac{1}{a}A_\mu\right) = -\sum_{l=0}^{d}\frac{\alpha_l(\mu)}{a}A_l. \tag{3.21}$$

We carry out the same analysis as in Section 3.1.2 to obtain

$$\left(I - \frac{1}{a}A_\mu\right)^p \approx \sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)\left(I - \frac{1}{a}A_{\mu_l}\right)^p, \quad \mu \in \mathcal{P}, \quad 1 \leq p \leq m, \tag{3.22}$$

where we recall that μ_l and $\lambda_l(\mu)$ are given by the EIM on $g(\vec{k}, \mu)$.

3.2.3. Approximation of the logarithm of the determinant of parametrized matrices

Replace $\left(I - \frac{1}{a}A_\mu\right)^p$ in the formula (3.17) by the right-hand side of (3.22) to obtain

$$\begin{aligned}
X_m g_\mu &\approx -\sum_{p=1}^{m-1}\frac{1}{p}\text{tr}\left(\sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)\left(I - \frac{1}{a}A_{\mu_l}\right)^p\right) \\
&= -\sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)\sum_{p=1}^{m-1}\frac{1}{p}\text{tr}\left(\left(I - \frac{1}{a}A_{\mu_l}\right)^p\right) \\
&= \sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)X_m g_{\mu_l}, \quad \mu \in \mathcal{P}.
\end{aligned} \tag{3.23}$$

Consider the following interpolation property:

Property 3.4 (see [22], Lem. 1). $\forall\, 1 \leq l \leq N^{\text{EIM}}, \forall \mu \in \mathcal{P}$,

$$I^{N^{\text{EIM}}}(g)(\vec{k}_l, \mu) = g(\vec{k}_l, \mu). \tag{3.24}$$

Notice that $\{\vec{k} \in [\![0; m]\!]^d$ such that $|\vec{k}| = 0\} = \{\vec{k}_0\}$, where $\vec{k}_0 := (0, 0, \cdots 0)$. Besides, $g(\vec{k}_0, \mu) = 1$ for all $\mu \in \mathcal{P}$. We impose the multi-indice $\vec{k}_0$ to be selected by the EIM in the offline stage, hence the EIM approximation of $g(\vec{k}_0, \mu)$ is exact for all $\mu \in \mathcal{P}$ by application of the interpolation Property 3.4. Hence, $\forall \mu \in \mathcal{P}$, $\sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)g(\vec{k}_0, \mu_l) = \sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu) = g(\vec{k}_0, \mu) = 1$, which enables us to write (3.23) as

$$X_m g_\mu + \mathcal{N}\log(\rho_M) = \sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)\left(X_m g_{\mu_l} + \mathcal{N}\log(\rho_M)\right), \quad \mu \in \mathcal{P}. \tag{3.25}$$

The convergence of $X_m g_\mu + \mathcal{N}\log(\rho_M)$ to $\log(\det(A_\mu))$ with respect to m, namely (3.20), suggests replacing $X_m g_{\mu_l} + \mathcal{N}\log(\rho_M)$ by $\log(\det(A_{\mu_l}))$ in (3.25) and defining

$$\mathcal{X}_\mu^{N^{\text{EIM}}} := \sum_{l=1}^{N^{\text{EIM}}}\lambda_l(\mu)\log(\det(A_{\mu_l})), \tag{3.26}$$

where $\mathcal{X}_\mu^{N^{\mathrm{EIM}}}$ is the obtained approximation of $\log(\det(A_\mu))$. Notice that we no longer need to compute ρ_M, and that any algorithm available to compute $\log(\det(A_{\mu_l}))$, $1 \le l \le N^{\mathrm{EIM}}$, even μ-dependent ones, can be used.

Notice that a key element of the section is the linearity of the function $g \mapsto X_m g_\mu$, where, from (3.8) and (3.17),

$$X_m g_\mu = - \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} g(c(\vec{k}), \mu) \left(\sum_{l=1}^{m} \frac{\mathrm{tr} \hat{T}_{\vec{k},l}}{l} \right). \tag{3.27}$$

3.3. Performance of the approximations

3.3.1. Reducibility

In an industrial context with large-scale computations and a constrained budget, the N^{EIM} in (3.13), (3.14), and (3.26) cannot be as large as we want. The success of any nonintrusive procedure will be assessed by the quality of the approximation within the given computation budget. If the approximation yields too large errors, the problem will be considered as nonreducible with the given procedure and the allocated computational budget. The proposed approximations have been motivated by the iterative schemes (3.1) and (3.17), which we recall are not required to be computed in practice. In (3.2), Ψ_0^{-1} can be seen as the best preconditioner uniformly on the parameter space. The problem can be efficiently reduced if this preconditioner is good in the sense that $\sup_{\mu \in \mathcal{P}} \|I - \Psi_0^{-1} A_\mu\|_2 = \rho \ll 1$, as can be seen in (3.6). In high parameter dimension cases, the existence of a good preconditioner is unlikely due to the curse of dimensionality, especially if the interval of variation of each parameter is large. We recall that we do not need to compute Ψ_0^{-1} and just need its existence. The success of the approximation will be assessed *a posteriori*, if a hidden low-rank structure exists, in the same fashion as other *a posteriori* reduced order methods, for instance in the snapshot POD if the eigenvalues of the correlation matrix decrease fast enough. In this context, at given computational budget, we compare our algorithm to some other nonintrusive procedures by computing the approximation errors with respect to reference values in Section 5.

3.3.2. Offline cost

Consider the approximations formulae (3.13), (3.14), and (3.26), that consist in the interpolation of respectively $A_{\mu_l}^{-1}$, $A_{\mu_l}^{-1} b$, and $\log(\det(A_{\mu_l}))$, $1 \le l \le N^{\mathrm{EIM}}$. The construction of these objects is inherent to any nonintrusive approximation method, where the high-fidelity model has to be solved a certain number of times to gather information to derive the approximation. The cost of computing $A_{\mu_l}^{-1}$, $A_{\mu_l}^{-1} b$ and $\log(\det(A_{\mu_l}))$, $1 \le l \le N^{\mathrm{EIM}}$, is then present in any nonintrusive method, and is not related to the offline part of the algorithm derived in the present work. The analysis boils down to assessing the cost of the computation of the coefficients $\lambda_l(\mu)$, $1 \le l \le N^{\mathrm{EIM}}$, in (3.13), (3.14), and (3.26). In our numerical applications, with d imposed by the form of the problem, we determine m_0 as the largest m such that $Q_{m,d} = \#\overline{\kappa}_{m,d}$ is lower than the computational budget. Then, the offline cost corresponds to the EIM applied to the function g on the sampled spaces $\overline{\kappa}_{m_0,d} \times \mathcal{P}_{\mathrm{sample}}$. In practice, since the computational budget is constrained (the largest value considered in our numerical experiments for $Q_{m_0,d}$ is 680), we have the opportunity to take a larger sampling of $\mathcal{P}$, which is desired anyway due to the possibly large dimension of $\mathcal{P}$. If the EIM is carried-out until all the $Q_{m_0,d}$ multi-indices in $\overline{\kappa}_{m_0,d}$ are selected, the algorithmic complexity is proportional to $Q_{m_0,d}^3 \times \#\mathcal{P}_{\mathrm{sample}}$: recall that in this case, $Q_{m_0,d}$ corresponds to the number of evaluations of the quantity of interest A_μ^{-1} or $\log(\det(A_\mu))$. In our numerical experiments, the offline stage of EIM with $\#\mathcal{P}_{\mathrm{sample}} = 10^6$ takes approximately the same time as the construction of the Design Of Experiment (DOE) using MaxProj when comparing with statistical methods, see Section 5.1 for more details. For instance, with $Q_{m_0,d} = 286$ ($m_0 = 2$ and $d = 10$), and $\#\mathcal{P}_{\mathrm{sample}} = 10^6$, both the construction of the DOE and the EIM take approximately 15 min.

Notice that in the classical use of EIM for order reduction of general nonlinear models where we want to approach the solution and/or operator, we need to evaluate the high-fidelity model $\#\mathcal{P}_{\mathrm{sample}}$ number of times: hence a large $\mathcal{P}_{\mathrm{sample}}$ is not a possile option. However, in the present work, the function g to approximate is

known on the complete set $\overline{\kappa}_{m_0,d} \times \mathcal{P}$ without solving the high-fidelity model, enabling the possibility of a large $\mathcal{P}_{\text{sample}}$.

4. CONVERGENCE OF THE APPROXIMATION

This section is organized as follows: Section 4.1 details the setting and notations, Section 4.2 states the main results, Section 4.3 gives the technical proofs, and comments are given in Section 4.4.

4.1. Setting

Recall the context of this work: we consider a parameter space $\mathcal{P}$, which is a compact subset of $\mathbb{R}^r$, and denote its Lebesgue mesure by $|\mathcal{P}|$. We also consider a family of matrices $\{A_\mu\}_{\mu \in \mathcal{P}} \subset \mathbb{R}^{\mathcal{N} \times \mathcal{N}}$. We look for approximations of quantities that can be obtained as limits of power algorithms applied to the matrices A_μ, denoted $\mathcal{L}_\mu$ (standing for "limit" for ease of reading): in the previous section, we considered the inverse matrix: $\mathcal{L}_\mu = A_\mu^{-1}$ and the log-det: $\mathcal{L}_\mu = \log(\det(A_\mu))$.

Let K be a bounded neighborhood of $\overline{\kappa}_{m,d}$ in $\mathbb{R}^d$, and denote $\mathcal{U} := L^2(K)$. We recall that $\overline{\kappa}_{m,d} = \left\{ \vec{k} \in [\![0;m]\!]^d \text{ such that } |\vec{k}| \le m \right\}$, and that $Q_{m,d} = \#\overline{\kappa}_{m,d} \le P_d(m)$, where $P_d(m)$ is a polynomial of degree d in m. Let $\mu \in \mathcal{P}$ and denote $g_\mu : \overline{\kappa}_{m,d} \ni \vec{k} \mapsto g_\mu(\vec{k}) := g(\vec{k},\mu) \in \mathbb{R}$. Consider the extension of g_μ from $\overline{\kappa}_{m,d}$ to K: $K \ni \vec{k} \mapsto g_\mu(\vec{k}) = \prod_{l=1}^d \alpha_l^{k_l}(\mu) \in \mathbb{R}$, from which we infer $g_\mu \in \mathcal{U} = L^2(K)$ – this point will be important later. We also suppose that α_l, $1 \le l \le d$, are continuous, which ensures the continuity of the functions $\mu \mapsto g(\vec{k},\mu)$ for all $\vec{k} \in K$.

We dispose of a sequence of linear applications $(X_m)_{m \in \mathbb{N}} \in L(\mathcal{U},V)$, where V is a Hilbert space of finite dimension s endowed with the scalar product $(.,.)_V$ and its associated norm $\|.\|_V := \sqrt{(.,.)_V}$ and where $L(\mathcal{U},V)$ denotes the space of linear applications from $\mathcal{U}$ to V. We suppose that the sequence $(X_m g_\mu)_{m \in \mathbb{N}}$ converges to $\mathcal{L}_\mu$ in the following sense: for all integer m and all $\mu \in \mathcal{P}$, $\|X_m g_\mu - \mathcal{L}_\mu\|_V \le C_1(m) \xrightarrow[m \to \infty]{} 0$. We precise here that even if $g_\mu \mapsto X_m g_\mu$ is linear, the dependence of the limit $\mathcal{L}_\mu$ with respect to g_μ is not necessarily linear. For the inverse matrix and the log-det applications, $\mu \mapsto \mathcal{L}_\mu$ is continuous due to the continuity of the α_l, and since $\mathcal{P}$ is a compact subset, $\sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu\|_V$ can be defined. Notice that since $\mu \mapsto X_m g_\mu$ is continuous for all m, the continuity of $\mu \mapsto \mathcal{L}_\mu$ can be obtained in the general case by assuming the uniform convergence of $(\mu \mapsto X_m g_\mu)$ to $(\mu \mapsto \mathcal{L}_\mu)$ with respect to m. We denote $\mathcal{C}^0(\mathcal{P},V)$, the Banach space of the continuous functions from $\mathcal{P}$ to V, endowed with the norm $\|w\|_{\mathcal{C}^0(\mathcal{P},V)} = \sup_{\mu \in \mathcal{P}} \|w(\mu)\|_V$.

For the inverse operators, the linear operator in g_μ is

$$\mathcal{U} \ni g_\mu \mapsto X_m g_\mu := \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} g(c(\vec{k}),\mu) \left(\hat{T}_{\vec{k},m} X_0 + \sum_{l=0}^{m-1} \hat{T}_{\vec{k},l} \Psi_0^{-1} \right) \in V := \mathbb{R}^{\mathcal{N} \times \mathcal{N}},$$

hence $s = \mathcal{N}^2$, the limit is $\mathcal{L}_\mu := A_\mu^{-1}$, and $C_1(m) = \epsilon_0 \rho^m$, see (3.6).

For the logarithm of the determinant, the linear operator in g_μ is

$$\mathcal{U} \ni g_\mu \mapsto X_m g_\mu := - \sum_{\vec{k} \in \overline{\kappa}_{m,d+1}} g(c(\vec{k}),\mu) \left(\sum_{l=1}^m \frac{\text{tr}\hat{T}_{\vec{k},l}}{l} \right) \in V := \mathbb{R},$$

hence $s = 1$, the limit is $\mathcal{L}_\mu := \log(\det(A_\mu)) - \mathcal{N} \log(\rho_M)$, and $C_1(m) = \mathcal{N} \frac{\rho_M}{\rho_0} \frac{\left(1 - \frac{\rho_0}{\rho_M}\right)^m}{m}$, see (3.20).

Consider the following EIM approximation of g:

$$I^{N^{\text{EIM}}}(g)(\vec{k},\mu) := \sum_{l=1}^{N^{\text{EIM}}} \lambda_l(\mu) g(\vec{k},\mu_l), \quad \vec{k} \in \overline{\kappa}_{m,d}, \quad \mu \in \mathcal{P},$$

TABLE 1. Definitions and properties resulting from the POD on the sets S.

Set	$S = \{g_{\mu_l}\}_{1 \le l \le N}$
Correlation operator	$C_{pq} = \left(g_{\mu_p}, g_{\mu_q}\right)_{\mathcal{U}}$
Eigenvalue problem	$\tau_n \xi_{n,p} = \frac{1}{N} \sum_{q=1}^{N} C_{pq} \xi_{n,q}$
POD modes	$\Phi_n = \frac{1}{\sqrt{N\tau_n}} \sum_{p=1}^{N} \xi_{n,p} g_{\mu_p}$
Eigenvalues property	$\tau_n = \frac{1}{N} \sum_{p=1}^{N} \left(g_{\mu_p}, \Phi_n\right)_{\mathcal{U}}^2$
Eigenfunctions orthonormality	$\sum_{p=1}^{N} \xi_{n,p} \xi_{m,p} = \delta_{n,m}$
POD modes orthonormality	$(\Phi_n, \Phi_m)_{\mathcal{U}} = \delta_{n,m}$

where we recall that $\lambda_l(\mu) = \sum_{l'=1}^{N^{\mathrm{EIM}}} \Delta_{l,l'} g(\vec{k}_{l'}, \mu)$, $\Delta = (F)^{-T}$ where $F_{l,l'} = g(\vec{k}_l, \mu_{l'})$, $1 \le l, l' \le N^{\mathrm{EIM}}$, where $\vec{k}_l$ and $\mu_{l'}$ are selected during the offline stage of EIM. We denote

$$\delta_{N^{\mathrm{EIM}}} = \left\| I^{N^{\mathrm{EIM}}}(g) - g \right\|_{L^2(\mathcal{P}, \mathcal{U})} := \sqrt{\int_{\mu \in \mathcal{P}} \left\| I^{N^{\mathrm{EIM}}}(g)(\cdot, \mu) - g(\cdot, \mu) \right\|_{\mathcal{U}}^2}, \tag{4.1}$$

which can be defined thanks to the compactness of $\mathcal{P}$ and the continuity of $\mu \mapsto g(\vec{k}, \mu)$ for all $\vec{k} \in K$, ensuring also the continuity of $\mu \mapsto \lambda_l(\mu)$, $1 \le l \le N^{\mathrm{EIM}}$, yielding the intregrability. Denote $\hat{Q}_{m,d} \le Q_{m,d}$, the rank of the matrix $\left(g(\vec{k}_i, \mu_j)\right)_{i,j}$, $1 \le i \le Q_{m,d}$, $1 \le j \le \#\mathcal{P}$. Owing to Property 3.4, $N^{\mathrm{EIM}} = \hat{Q}_{m,d}$ implies that the EIM approximation is exact on the whole domain $\overline{\kappa}_{m,d} \times \mathcal{P}$. Hence, we now consider values for N^{EIM} smaller than $\hat{Q}_{m,d}$. For ease of reading, we set $N := N^{\mathrm{EIM}}$, keeping in mind the dependency of N in m.

In what follows, we denote by $(.,.)_{\mathcal{U}}$ the scalar product on $\mathcal{U}$ and $\|.\|_{\mathcal{U}}$ its associated norm. The corresponding inner product is the L^2- one. As explained at the beginning of the section, $g_\mu \in \mathcal{U}$, for all $\mu \in \mathcal{P}$. Denote the set $S = \{g_{\mu_l}\}_{1 \le l \le N} \subset \mathcal{U}$ where the μ_l are the parameter values selected by the EIM on g. We apply the POD technique to the set S, see Table 1 for the obtained properties and [4, 30] for more details and justifications.

The approximation of $\mathcal{L}_\mu$, denoted $\mathcal{X}_\mu^N$, is defined as

$$\mathcal{X}_\mu^N := \sum_{n=1}^{N} \lambda_n(\mu) \mathcal{L}_{\mu_n}. \tag{4.2}$$

4.2. Main results

In this section, we give two different bounds for the error made by the approximation $\mathcal{X}_\mu^N$ of $\mathcal{L}_\mu$ the first one involves a rather abstract vector space, the second one makes use of the relation between the functions g_μ, on which the EIM approximation is carried out, and the approximated object $\mathcal{L}_\mu$, through the iterative schemes $X_m g_\mu$.

Define $Z_N := \frac{1}{N} \sum_{n=1}^{N} \frac{\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2}{\frac{|\mathcal{P}|}{N} \sum_{p=1}^{N} (g_{\mu_p}, \Phi_n)_{\mathcal{U}}^2}$: in the quotient, the denominator is an approximation of the numerator, leading to the boundedness of $(Z_N)_N$. Define also $\mathcal{S}^{sN}$ the smallest sN-dimensional subspace of $\mathcal{C}^0(\mathcal{P}, V)$ containing the image of the application $v \mapsto \mathcal{J}^N v$, defined by $\forall \mu \in \mathcal{P}$, $(\mathcal{J}^N v)(\mu) := \sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{N} G_{np}^{-1} v(\mu_p)$, where $G_{np} = (g_{\mu_n}, \Phi_p)_{\mathcal{U}} \in \mathbb{R}^{N \times N}$ is an invertible matrix. The boundedness of $(Z_N)_N$, the dimension of $\mathcal{S}^{sN}$ and the invertibility of G will be justified in Section 4.3.

Proposition 4.1. *For any integer m and $1 \le N < \hat{Q}_{m,d}$,*

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \|\mathcal{X}_\mu^N - \mathcal{L}_\mu\|_V^2 \le 4 \left(1 + N^2 Z_N\right) \left(\theta_{\mathcal{L}}^{sN}\right)^2 + \frac{8}{|\mathcal{P}|} \sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu\|_V^2 \frac{N \delta_N^2}{\tau_N}, \tag{4.3}$$

where

$$\theta_{\mathcal{L}}^{sN} := \inf_{\varphi \in \mathcal{S}^{sN}} \sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu - \varphi(\mu)\|_V. \tag{4.4}$$

In the case $N = \hat{Q}_{m,d}$ where the EIM approximation is exact: for any integer m

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \|\mathcal{X}_\mu^{\hat{Q}_{m,d}} - \mathcal{L}_\mu\|_V^2 \leq 4 \left(1 + \hat{Q}_{m,d}^2 Z_{\hat{Q}_{m,d}}\right) \left(\theta_{\mathcal{L}}^{s\hat{Q}_{m,d}}\right)^2. \tag{4.5}$$

Proposition 4.2. *For any integer m and $1 \leq N < \hat{Q}_{m,d}$,*

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \|\mathcal{X}_\mu^N - \mathcal{L}_\mu\|_V^2 \leq 3C_1^2(m) \left(1 + 2N^2 Z_N + 8 \frac{1}{|\mathcal{P}|} \frac{N\delta_N^2}{\tau_N}\right) + 3\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \left\|X_m \left(I^N(g_\mu) - g_\mu\right)\right\|_V^2. \tag{4.6}$$

In the case $N = \hat{Q}_{m,d}$ where the EIM approximation is exact: for any integer m

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \|\mathcal{X}_\mu^{\hat{Q}_{m,d}} - \mathcal{L}_\mu\|_V^2 \leq 3C_1^2(m) \left(1 + 2\hat{Q}_{m,d}^2 Z_{\hat{Q}_{m,d}}\right). \tag{4.7}$$

Remark 4.3. The bounds in (4.3) and (4.6) involve $\frac{N\delta_N^2}{\tau_N}$ and $\left\|X_m \left(I^N(g_\mu) - g_\mu\right)\right\|_V^2$, which are difficult to describe: on the one hand the asymptotic behavior of $\frac{N\delta_N^2}{\tau_N}$ exibits an indeterminate form, and on the other hand the operator norm of X_m is hard to estimate. However, thanks to the interpolation property of the EIM, we know that $\delta_{\hat{Q}_{m,d}} = 0$ and $I^{\hat{Q}_{m,d}}(g_\mu) = g_\mu$, while $\tau_{\hat{Q}_{m,d}} > 0$: sharper upper bounds are derived in this particular case.

In (4.5), $Z_{\hat{Q}_{m,d}}$ is bounded, and $\theta_{\mathcal{L}}^{s\hat{Q}_{m,d}}$ is related to a certain Kolmogorov width, which are usually assumed to compensate for exponential or polynomial growth in approximation problems; in our case, we only need to assume the convergence of $\hat{Q}_{m,d}\theta_{\mathcal{L}}^{s\hat{Q}_{m,d}}$. This is thoroughly commented in Section 4.4. In (4.7), the convergence is ensured by the properties of the considered power algorithm through $C_1(m)$, as we explicit in the following corollary.

Corollary 4.4. *In the case $N = \hat{Q}_{m,d}$ where the EIM approximation is exact, the bound of Proposition 4.2 is*

– *for the inverse matrix:*

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \|\mathcal{X}_\mu^{\hat{Q}_{m,d}} - A_\mu^{-1}\|_V^2 \leq 3\epsilon_0^2 \rho^{2m} \left(1 + 2P_d^2(m) Z_{\hat{Q}_{m,d}}\right), \tag{4.8}$$

– *for the log-det:*

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} |\mathcal{X}_\mu^{\hat{Q}_{m,d}} - \log(\det(A_\mu))|^2 \leq 3\frac{\mathcal{N}^2}{m^2} \frac{\rho_M^2}{\rho_0^2} \left(1 - \frac{\rho_0}{\rho_M}\right)^{2m} \left(1 + 2P_d^2(m) Z_{\hat{Q}_{m,d}}\right), \tag{4.9}$$

where we recall that $P_d(m)$ is a polynomial of degree d in m, such that $\hat{Q}_{m,d} \leq Q_{m,d} \leq P_d(m)$. The approximation converges under the condition that $\rho < 1$ for the inverse matrix, and that $1 - \frac{\rho_0}{\rho_M} < 1$ for the log-det.

Remark 4.5 (Reducibility)**.** In the case of the inverse matrix, the bound in (4.8) converges under the strong assumption that $\rho < 1$, where we recall that $\rho = \sup_{\mu \in \mathcal{P}} \|I - \Psi_0^{-1} A_\mu\|_2$. The strength of the assumption lies in the existence of a good preconditioner Ψ_0 uniformly on the possibly large dimensional parameter space $\mathcal{P}$, which we have related to the reducibility of the problem at hand in Section 3.3.1.

4.3. Technical proofs

In this section, we start by giving two results on the POD basis $(\Phi_n)_{n\in\mathbb{N}}$: Intermediate Results 4.6 and 4.8, from which we derive the proofs of Propositions 4.1 and 4.2.

Intermediate Result 4.6. *The matrix* $G_{np} = (g_{\mu_n}, \Phi_p)_{\mathcal{U}} \in \mathbb{R}^{N\times N}$ *is invertible, and* $\forall \vec{r} \in \mathbb{R}^N$, $\forall \vec{v} := (v_n)_{1\leq n\leq N}$ *with* $v_n \in V$, *there holds*

$$\left\| \vec{r}^{\;t} \cdot (G^{-1}\vec{v}) \right\|_V^2 \leq \left(\sum_{n=1}^{N} \|v_n\|_V^2 \right) \left(\sum_{n=1}^{N} \frac{r_n^2}{\tau_n} \right). \tag{4.10}$$

Proof of Intermediate Result 4.6. The family $(g_{\mu_n})_{1\leq n\leq N}$ is free over $\mathcal{U}$. Indeed, let $\vec{a} \in \mathbb{R}^N$ such that $\sum_{n=1}^{N} a_n g_{\mu_n} = 0$. The equality holds in particular for the indices $\vec{k}_l \in \overline{\kappa}_{m,d}$ selected by EIM: $\forall \; 1 \leq l \leq N$, $\sum_{n=1}^{N} a_n g(\mu_n, \vec{k}_l) = 0$. By construction of the EIM, the matrix $(g(\vec{k}_l, \mu_n))_{1\leq l,n\leq N}$ is invertible (see [8], Lem. 2.2), in particular the rows of this matrix form a free family. This entails that all the a_n are zero, which proves that the family $(g_{\mu_n})_{1\leq n\leq N}$ is free over $\mathcal{U}$. Then, thanks to the orthonormality of the basis $(\Phi_n)_{n\in\mathbb{N}}$, $g_{\mu_n} = \sum_{p=1}^{N} (g_{\mu_n}, \Phi_p)_{\mathcal{U}} \Phi_p = \sum_{p=1}^{N} G_{np}\Phi_p$, for all $1 \leq n \leq N$, which means that G is the change of basis matrix from $(\Phi_n)_{1\leq n\leq N}$ to $(g_{\mu_p})_{1\leq p\leq N}$, and is therefore invertible.

Hence, $\Phi_n = \sum_{p=1}^{N} G_{np}^{-1} g_{\mu_p}$. From the definition of the POD modes Φ_n (see Tab. 1) and due to the fact that $(g_{\mu_i})_{1\leq i\leq N}$ is a set of linear independent vectors, we obtain $G_{np}^{-1} = \frac{1}{\sqrt{N\tau_n}}\xi_{n,p}$ for all $1 \leq n, p \leq N$. Then, using the eigenfunctions orthonormality (see Tab. 1),

$$\sum_{p=1}^{N} \left(G_{np}^{-1}\right)^2 = \frac{1}{N\tau_n} \sum_{p=1}^{N} \xi_{n,p}^2 = \frac{1}{N\tau_n}. \tag{4.11}$$

There holds

$$\left\| \vec{r}^{\;t} \cdot (G^{-1}\vec{v}) \right\|_V^2 = \left(\left\| \sum_{n=1}^{N} r_n \sum_{p=1}^{N} G_{np}^{-1} v_p \right\|_V \right)^2 \tag{4.12a}$$

$$\leq \left(\sum_{n=1}^{N} |r_n| \left\| \sum_{p=1}^{N} G_{np}^{-1} v_p \right\|_V \right)^2 \tag{4.12b}$$

$$\leq N \sum_{n=1}^{N} r_n^2 \left\| \sum_{p=1}^{N} G_{np}^{-1} v_p \right\|_V^2 \tag{4.12c}$$

$$\leq N \sum_{n=1}^{N} r_n^2 \left(\sum_{p=1}^{N} |G_{np}^{-1}| \, \|v_p\|_V \right)^2 \tag{4.12d}$$

$$\leq N \sum_{n=1}^{N} r_n^2 \left(\sum_{p=1}^{N} \left(G_{np}^{-1}\right)^2 \right) \left(\sum_{p=1}^{N} \|v_p\|_V^2 \right) \tag{4.12e}$$

$$\leq \left(\sum_{n=1}^{N} \|v_n\|_V^2 \right) \left(\sum_{n=1}^{N} \frac{r_n^2}{\tau_n} \right), \tag{4.12f}$$

where the Jensen inequality is applied to the square function between (4.12b) and (4.12c), the Cauchy–Schwarz inequality is applied between (4.12d) and (4.12e), and where (4.12f) is obtained from (4.12e) using (4.11), which ends the proof. $\qquad\square$

Remark 4.7. In the proof of Intermediate result 4.6, we could have simply introduced an orthonormal basis $(\Phi_n)_{1 \leq n \leq N}$, obtained for instance from a Gram-Schmidt orthonormalization of the family g_{μ_l} $1 \leq l \leq N$, and used $\sum_{p=1}^{N} \left(G_{np}^{-1} \right)^2 \leq \| G^{-1} \|_F^2$, where $\| G^{-1} \|_F$ is the Frobenius norm of G^{-1}, depending on N. However, doing so would not have yielded a bound with an explicit dependence on N.

Intermediate Result 4.8.
$$\int_{\mu \in \mathcal{P}} \left\| \Pi_N^{\Phi} I^N(g_\mu) - \Pi_N^{\Phi} g_\mu \right\|_{\mathcal{U}}^2 \leq 4\delta_N^2,$$

where Π_N^{Φ} is the orthogonal projection operator onto the subspace $\underset{1 \leq n \leq N}{\mathrm{Span}} (\Phi_n)$ and where we recall that δ_N quantifies the EIM approximation error, see (4.1).

Proof of Intermediate Result 4.8.
$$\Pi_N^{\Phi} I^N(g_\mu) - \Pi_N^{\Phi} g_\mu = \Pi_N^{\Phi} I^N(g_\mu) - g_\mu + g_\mu - \Pi_N^{\Phi} g_\mu.$$

Recall that $\underset{1 \leq n \leq N}{\mathrm{Span}} (\Phi_n) = \underset{1 \leq n \leq N}{\mathrm{Span}} (g_{\mu_n})$ from Intermediate Result 4.6, providing $\Pi_N^{\Phi} I^N(g_\mu) = I^N(g_\mu)$. Using the triangular inequality hence yields

$$
\begin{aligned}
\int_{\mu \in \mathcal{P}} \left\| \Pi_N^{\Phi} I^N(g_\mu) - \Pi_N^{\Phi} g_\mu \right\|_{\mathcal{U}}^2 &\leq 2 \int_{\mu \in \mathcal{P}} \left\| I^N(g_\mu) - g_\mu \right\|_{\mathcal{U}}^2 + 2 \int_{\mu \in \mathcal{P}} \left\| g_\mu - \Pi_N^{\Phi} g_\mu \right\|_{\mathcal{U}}^2 \\
&= 2 \int_{\mu \in \mathcal{P}} \left\| I^N(g_\mu) - g_\mu \right\|_{\mathcal{U}}^2 + 2 \int_{\mu \in \mathcal{P}} \inf_{v \in \underset{1 \leq n \leq N}{\mathrm{Span}} (\Phi_n)} \left\| g_\mu - v \right\|_{\mathcal{U}}^2 \\
&\leq 4 \int_{\mu \in \mathcal{P}} \left\| I^N(g_\mu) - g_\mu \right\|_{\mathcal{U}}^2 \\
&\leq 4\delta_N^2,
\end{aligned}
$$

which ends the proof. $\qquad\square$

Proof of Proposition 4.1. First, we recall the definition of the application $\mathcal{C}^0(\mathcal{P}, V) \ni v \mapsto \mathcal{J}^N v \in \mathcal{C}^0(\mathcal{P}, V)$ such that $\forall \mu \in \mathcal{P}$, $(\mathcal{J}^N v)(\mu) := \sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{N} G_{np}^{-1} v(\mu_p)$, and of the subspace $\mathcal{S}^{sN}$: it is the smallest subspace of $\mathcal{C}^0(\mathcal{P}, V)$ containing the image of $\mathcal{J}^N$. To see that the dimension of $\mathcal{S}^{sN}$ is sN, denote $(e_i)_{1 \leq i \leq s}$ a basis of V; any element w of V can be expressed in this basis as follows: $w = \sum_{i=1}^{s} \eta_i(w) e_i$. Then, for all $v \in \mathcal{C}^0(\mathcal{P}, V)$, there holds $(\mathcal{J}^N v)(\mu) = \sum_{p=1}^{N} \sum_{i=1}^{s} (\eta_i(v(\mu_p))) \left(\sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} G_{np}^{-1} e_i \right)$, where $\eta_i(v(\mu_p)) \in \mathbb{R}$ and $\sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} G_{np}^{-1} e_i \in \mathcal{C}^0(\mathcal{P}, V)$ and is independent of v. This proves that the family $\left(\left(\sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} G_{np}^{-1} \right) e_i \right)_{1 \leq p \leq N, \, 1 \leq i \leq s}$ is composed of spanning vectors of $\mathcal{S}^{sN}$. We conclude by noticing that this family is free in $\mathcal{C}^0(\mathcal{P}, \mathbb{R}) \times V \subset \mathcal{C}^0(\mathcal{P}, V)$: $(e_i)_{1 \leq i \leq s}$ is a basis of V, and let $\omega \in \mathbb{R}^N$ such that $\sum_{p=1}^{N} \omega_p \left(\sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} G_{np}^{-1} \right) = 0$ in $\mathcal{C}^0(\mathcal{P}, \mathbb{R})$. In particular, $\forall 1 \leq q \leq N$, $\sum_{p=1}^{N} \omega_p \left(\sum_{n=1}^{N} (g_{\mu_q}, \Phi_n)_{\mathcal{U}} G_{np}^{-1} \right) = \sum_{p=1}^{N} \omega_p \left(G G^{-1} \right)_{qp} = \omega_q = 0$.

We now go back to the control of

$$\int_{\mu \in \mathcal{P}} \left\| \mathcal{X}_\mu^N - \mathcal{L}_\mu \right\|_V^2 = \int_{\mu \in \mathcal{P}} \left\| \sum_{n=1}^{N} \lambda_n(\mu) \mathcal{L}_{\mu_n} - \mathcal{L}_\mu \right\|_V^2.$$

Denote $\vec{\lambda}(\mu) := (\lambda_n(\mu))_{1 \le n \le N}$, $\vec{\mathcal{L}} := (\mathcal{L}_{\mu_n})_{1 \le n \le N}$ and $\vec{h}(\mu) = ((g_\mu, \Phi_n)_{\mathcal{U}})_{1 \le n \le N}$. With these notations, $(\mathcal{J}^N \mathcal{L})(\mu) := \vec{h}^t(\mu) \cdot G^{-1} \vec{\mathcal{L}}$. Then,

$$\int_{\mu \in \mathcal{P}} \left\| \sum_{n=1}^{N} \lambda_n(\mu) \mathcal{L}_{\mu_n} - \mathcal{L}_\mu \right\|_V^2 = \int_{\mu \in \mathcal{P}} \left\| \vec{\lambda}^t(\mu) \cdot \vec{\mathcal{L}} - \mathcal{L}_\mu \right\|_V^2 \tag{4.13a}$$

$$\le 2 \int_{\mu \in \mathcal{P}} \left\| \left(G^t \vec{\lambda}(\mu) - \vec{h}(\mu) \right)^t \cdot \left(G^{-1} \vec{\mathcal{L}} \right) \right\|_V^2 + 2 \int_{\mu \in \mathcal{P}} \left\| (\mathcal{J}^N \mathcal{L})(\mu) - \mathcal{L}_\mu \right\|_V^2. \tag{4.13b}$$

We now control the two terms in (4.13b):

– (first term in (4.13b))

$$2 \int_{\mu \in \mathcal{P}} \left\| \left(G^t \vec{\lambda}(\mu) - \vec{h}(\mu) \right)^t \cdot \left(G^{-1} \vec{\mathcal{L}} \right) \right\|_V^2 \le 2 \left(\sum_{n=1}^{N} \mathcal{L}_{\mu_n}^2 \right) \int_{\mu \in \mathcal{P}} \sum_{n=1}^{N} \frac{1}{\tau_n} \left(G^t \vec{\lambda}(\mu) - \vec{h}(\mu) \right)_n^2 \tag{4.14a}$$

$$\le 2 \sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu\|_V^2 N \int_{\mu \in \mathcal{P}} \sum_{n=1}^{N} \frac{1}{\tau_n} \left(\sum_{p=1}^{N} \lambda_p(\mu) g_{\mu_p} - g_\mu, \Phi_n \right)_{\mathcal{U}}^2 \tag{4.14b}$$

$$\le 2 \sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu\|_V^2 \frac{N}{\tau_N} \int_{\mu \in \mathcal{P}} \left\| \Pi_N^\Phi \left(I^N g_\mu - g_\mu \right) \right\|_{\mathcal{U}}^2 \tag{4.14c}$$

$$\le 8 \sup_{\mu \in \mathcal{P}} \|\mathcal{L}_\mu\|_V^2 \frac{N \delta_N^2}{\tau_N}, \tag{4.14d}$$

where we applied Intermediate Result 4.6 to $\vec{r} = G^t \vec{\lambda}(\mu) - \vec{h}(\mu)$ and $\vec{v} = \vec{\mathcal{L}}$ in (4.14a) and Intermediate Result 4.8 between (4.14c) and (4.14d).

– (second term in (4.13b)) For all $v \in \mathcal{C}^0(\mathcal{P}, V)$, the application $(\mathcal{J}^N v)(\mu)$ defines an interpolation, since $\forall \, 1 \le q \le N$,

$$\left(\mathcal{J}^N v \right) (\mu_q) = \sum_{n=1}^{N} (g_{\mu_q}, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{N} G_{np}^{-1} v(\mu_p) = \sum_{p=1}^{N} \left(G G^{-1} \right)_{qp} v(\mu_p) = v(\mu_q). \tag{4.15}$$

Moreover, $\mathcal{J}^N$ is a linear projector onto $\mathcal{S}^{sN}$ since using (4.15), for all $v \in \mathcal{C}^0(\mathcal{P}, V)$,

$$\left(\mathcal{J}^N \mathcal{J}^N v \right) (\mu) = \sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{N} G_{np}^{-1} (\mathcal{J}^N v)(\mu_p) = \sum_{n=1}^{N} (g_\mu, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{N} G_{np}^{-1} v(\mu_p) = \left(\mathcal{J}^N v \right) (\mu). \tag{4.16}$$

Now denote $\varphi_0 = \underset{\varphi \in \mathcal{S}^{sN}}{\operatorname{arginf}} \|\varphi - \mathcal{L}\|_{\mathcal{C}^0(\mathcal{P}, V)}$. Since $\mathcal{J}^N$ is a projector, $\varphi_0 = \mathcal{J}^N \varphi_0$ in $\mathcal{C}^0(\mathcal{P}, V)$. We control the second term in (4.13b) with

$$2 \int_{\mu \in \mathcal{P}} \left\| (\mathcal{J}^N \mathcal{L})(\mu) - \mathcal{L}_\mu \right\|_V^2 = 2 \left\| \mathcal{J}^N \mathcal{L} - \mathcal{L} \right\|_{L^2(\mathcal{P}, V)}^2 \le 4 \left\| \mathcal{J}^N \mathcal{L} - \varphi_0 \right\|_{L^2(\mathcal{P}, V)}^2 + 4 \left\| \varphi_0 - \mathcal{L} \right\|_{L^2(\mathcal{P}, V)}^2$$

$$\le 4 \left\| \mathcal{J}^N \left(\mathcal{L} - \varphi_0 \right) \right\|_{L^2(\mathcal{P}, V)}^2 + 4 |\mathcal{P}| \left\| \varphi_0 - \mathcal{L} \right\|_{\mathcal{C}^0(\mathcal{P}, V)}^2$$

$$= 4 \int_{\mu \in \mathcal{P}} \left\| \vec{h}^t(\mu) \cdot G^{-1} \left(\overrightarrow{\mathcal{L} - \varphi_0} \right) \right\|_V^2 + 4 |\mathcal{P}| \left(\theta_{\mathcal{L}}^{sN} \right)^2. \tag{4.17}$$

Using Intermediate Result 4.6 with $\vec{r} = \vec{h}(\mu)$, there holds,

$$4 \int_{\mu \in \mathcal{P}} \left\| \vec{h}^t(\mu) \cdot G^{-1} \left(\overrightarrow{\mathcal{L} - \varphi_0} \right) \right\|_V^2 \leq 4 \int_{\mu \in \mathcal{P}} \left(\sum_{n=1}^{N} \| \mathcal{L}_{\mu_n} - \varphi_0(\mu_n) \|_V^2 \right) \left(\sum_{n=1}^{N} \frac{(g_\mu, \Phi_n)_{\mathcal{U}}^2}{\tau_n} \right) \tag{4.18a}$$

$$\leq 4N \sup_{\mu \in \mathcal{P}} \| \mathcal{L}_\mu - \varphi_0(\mu) \|_V^2 \left(\sum_{n=1}^{N} \frac{\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2}{\tau_n} \right) \tag{4.18b}$$

$$\leq 4|\mathcal{P}| N^2 Z_N \left(\theta_{\mathcal{L}}^{sN} \right)^2, \tag{4.18c}$$

where we recall that $Z_N = \dfrac{1}{N} \displaystyle\sum_{n=1}^{N} \dfrac{\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2}{\frac{|\mathcal{P}|}{N} \sum_{p=1}^{N} (g_{\mu_p}, \Phi_n)_{\mathcal{U}}^2}$. We recall that the considered POD has been carried-out on the discrete set $S = \{g_{\mu_l}\}_{1 \leq l \leq N}$, see Table 1 for the definition and the properties of this POD. Consider $\tilde{S} = \{g_\mu\}_{\mu \in \mathcal{P}}$: a second POD applied on this set leads to POD modes denoted $\tilde{\Phi}_n$ and eigenvalues given by $\tilde{\tau}_n := \frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \left(g_\mu, \tilde{\Phi}_n \right)_{\mathcal{U}}^2, 1 \leq n \leq N$. The POD decompositions on S and $\tilde{S}$ are asymptotically equal when N tends to infinity (this corresponds to the case where the dimension of subspace spanned by the elements of $\tilde{S}$ is infinite), leading to: $\forall n \geq 1, \Phi_n \underset{N \to \infty}{\to} \tilde{\Phi}_n$ and $\tau_n \underset{N \to \infty}{\to} \tilde{\tau}_n$. Hence, we infer $\forall n \geq 1, \dfrac{\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2}{\frac{|\mathcal{P}|}{N} \sum_{p=1}^{N} (g_{\mu_p}, \Phi_n)_{\mathcal{U}}^2} \underset{N \to \infty}{\to} 1$, leading to $Z_N \underset{N \to \infty}{\to} 1$, then Z_N is bounded.

We now conclude the proof from the control of the terms in (4.13b) and by noting that $\delta_{\hat{Q}_{m,d}} = 0$ owing to Property 3.4, and $\tau_{\hat{Q}_{m,d}} > 0$. □

Remark 4.9 (Boundedness of $(Z_N)_N$). Another way to control the bound of $(Z_N)_N$ is to recognize a Riemann sum in the denominator of the quotient in the definition of Z_N, for which the sampling μ_p is selected by an EIM on g. The problem of defining the best sample of points to construct an interpolation is complex and is in general not solved, but the sample provided by EIM is competitive compared to situations where the best behavior is known, see the numerical illustrations in [22]. In pratice, in [22] and in Figure 4, we observe the points selected by the EIM to be distributed quite regularly in the parameter space (in particular, the EIM cannot select twice the same point). Construct a Voronoi tesselation of $\mathcal{P}$ from this set of points, and denote v_N^p the volume of the cells. Denote $M_N := \frac{N}{|\mathcal{P}|} \sup_{1 \leq p \leq N} v_N^p$, since $\frac{|\mathcal{P}|}{N}$ is the mean volume of the cells, the assumption of regular distribution for the EIM points leads to M_N is close to 1 for all N. Then, $Z_N \leq M_N \dfrac{1}{N} \displaystyle\sum_{n=1}^{N} \dfrac{\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2}{\sum_{p=1}^{N} v_p^N (g_{\mu_p}, \Phi_n)_{\mathcal{U}}^2}$, where $\displaystyle\sum_{p=1}^{N} v_p^N (g_{\mu_p}, \Phi_n)_{\mathcal{U}}^2$ is a Riemann sum converging to the integral $\displaystyle\int_{\mu \in \mathcal{P}} (g_\mu, \Phi_n)_{\mathcal{U}}^2$ as N tends to infinity.

Proof of Proposition 4.2. Using the triangular inequality, there holds

$$\int_{\mu \in \mathcal{P}} \| \mathcal{X}_\mu^N - \mathcal{L}_\mu \|_V^2 \leq 3 \int_{\mu \in \mathcal{P}} \| \mathcal{L}_\mu - X_m g_\mu \|_V^2 + 3 \int_{\mu \in \mathcal{P}} \left\| \sum_{n=1}^{N} \lambda_n(\mu) \left(\mathcal{L}_{\mu_n} - X_m g_{\mu_n} \right) \right\|_V^2$$

$$+ 3 \int_{\mu \in \mathcal{P}} \left\| X_m \left(I^N(g_\mu) - g_\mu \right) \right\|_V^2, \tag{4.19}$$

where we recall that $X_m g_\mu$ is the mth term of the considered power algorithm at parameter value μ, converging to the quantity of interest $\mathcal{L}_\mu$, and the EIM approximation of g is $I^N(g)(\vec{k}, \mu) = \sum_{n=1}^{N} \lambda_n(\mu) g(\vec{k}, \mu_n)$. The

second term in the right-hand side can be controlled in the same fashion as in the proof of Proposition 4.1: replacing $\mathcal{L}_\mu$ with $\mathcal{L}_\mu - X_m g_\mu$, there holds, denoting $\overrightarrow{\mathcal{L} - X_m g} = (\mathcal{L}_{\mu_n} - X_m g_{\mu_n})_{1 \leq n \leq N}$,

$$3 \int_{\mu \in \mathcal{P}} \left\| \sum_{n=1}^N \lambda_n(\mu) \left(\mathcal{L}_{\mu_n} - X_m g_{\mu_n} \right) \right\|_V^2 \leq 6 \int_{\mu \in \mathcal{P}} \left\| \left(G^t \vec{\lambda}(\mu) - \vec{h}(\mu) \right)^t \cdot \left(G^{-1} \left(\overrightarrow{\mathcal{L} - X_m g} \right) \right) \right\|_V^2 \tag{4.20a}$$

$$+ 6 \int_{\mu \in \mathcal{P}} \left\| \vec{h}^t(\mu) \cdot \left(G^{-1} \left(\overrightarrow{\mathcal{L} - X_m g} \right) \right) \right\|_V^2 \tag{4.20b}$$

$$\leq 24 C_1^2(m) \frac{N \delta_N^2}{\tau_N} + 6 |\mathcal{P}| C_1^2(m) N^2 Z_N, \tag{4.20c}$$

where we recall that $C_1(m)$ is a bound for the approximation of the considered power matrix algorithm, δ_N is the EIM approximation error, $(Z_N)_N$ is a bounded sequence, and τ_N denotes the Nth eigenvalue of the considered POD, see Table 1. The control of the first term in the right-hand side of (4.20b) was obtained in (4.14d) replacing $\vec{\mathcal{L}}$ by $\overrightarrow{\mathcal{L} - X_m g}$, and the control of the second term was obtained in (4.18c) replacing φ_0 by $X_m g$. The proof is ended by noting that $\delta_{\hat{Q}_{m,d}} = 0$ and $\forall \mu \in \mathcal{P}$, $I^{\hat{Q}_{m,d}}(g_\mu) = g_\mu$ owing to Property 3.4. $\qquad \square$

4.4. Comments

We recall the practical results of Section 4.2 stated in Corollary 4.4: the nonintrusive approximations for the inverse matrix (3.13) and for the log-det (3.26) are convergent with respect to the number of evaluations N^{EIM} of the quantity of interest.

Consider now F a compact subset of $\mathcal{C}^0(\mathcal{P}, V)$ containing $\mathcal{L}$. The second bound in Proposition 4.1 can be weakened to the form: for all integer m,

$$\frac{1}{|\mathcal{P}|} \int_{\mu \in \mathcal{P}} \| \mathcal{X}_\mu^{\hat{Q}_{m,d}} - \mathcal{L}_\mu \|_V^2 \leq 4 \left(1 + \hat{Q}_{m,d}^2 Z_{\hat{Q}_{m,d}} \right) \eta_{\mathcal{S}^{s\hat{Q}_{m,d}}}^2, \tag{4.21}$$

where

$$\eta_{\mathcal{S}^{s\hat{Q}_{m,d}}} := \sup_{v \in F} \inf_{\varphi \in \mathcal{S}^{s\hat{Q}_{m,d}}} \| v - \varphi \|_{\mathcal{C}^0(\mathcal{P}, V)}, \tag{4.22}$$

which is related to the following Kolmogorov $s\hat{Q}_{m,d}$-width:

$$d_{s\hat{Q}_{m,d}}(F, \mathcal{C}^0(\mathcal{P}, V)) = \inf_{F^{s\hat{Q}_{m,d}} \subset \mathcal{C}^0(\mathcal{P}, V)} \sup_{v \in F} \inf_{\varphi \in F^{s\hat{Q}_{m,d}}} \| v - \varphi \|_{\mathcal{C}^0(\mathcal{P}, V)} = \inf_{F^{s\hat{Q}_{m,d}} \subset \mathcal{C}^0(\mathcal{P}, V)} \eta_{F^{s\hat{Q}_{m,d}}}. \tag{4.23}$$

For the need of the proof, we considered the snapshots POD on the set $(g_{\mu_l})_{1 \leq l \leq \hat{Q}_{m,d}}$, with values μ_n selected by a first EIM on g, which lead to a fixed projector $(\mathcal{J}^{\hat{Q}_{m,d}} v)(\mu) = \sum_{n=1}^{\hat{Q}_{m,d}} (g_\mu, \Phi_n)_{\mathcal{U}} \sum_{p=1}^{\hat{Q}_{m,d}} G_{np}^{-1} v(\mu_p)$ for $v \in \mathcal{C}^0(\mathcal{P}, V)$, and therefore a fixed subspace $\mathcal{S}^{s\hat{Q}_{m,d}} \subset \mathcal{C}^0(\mathcal{P}, V)$, instead of the optimal subspace $F^{s\hat{Q}_{m,d}}$ in (4.23).

In [21], upper bounds for the EIM error have been derived, for polynomial and exponential decay rates of the Kolmogorov n-width $d_n(\{g_\mu, \mu \in \mathcal{P}\}, \mathcal{U})$. In Proposition 4.1 are made explicit the dependences on the EIM upper bound δ_N and on $\theta_{\mathcal{L}}^{sN}$, which is related to the approximation of $\mu \mapsto \mathcal{L}_\mu$ in $\mathcal{C}^0(\mathcal{P}, V)$, not to the EIM approximation of g_μ in $\mathcal{U}$.

The convergence of the upper bounds in Proposition 4.1 are difficult to observe in practice, due to difficulty of the numerical estimation of δ_N and $\theta_{\mathcal{L}}^{sN}$. However, the convergence of the upper bound in (4.5) seams reasonnable since, in reducible cases, the convergence of $N \theta_{\mathcal{L}}^{sN}$ is a mild assumption when $\theta_{\mathcal{L}}^{sN}$ is replaced by the Kolmogorov sN-width $d_{sN}(F, \mathcal{C}^0(\mathcal{P}, V))$. In our numerical experiments, we observed that the EIM provides reasonable approximation errors only in the case where $N = \hat{Q}_{m,d} = Q_{m,d}$, probably due to the particular form of $\overline{\kappa}_{m,d}$, which is a discrete set of multi-indices, not a discrete sampling of some continuous variable: the elements

$\vec{k}$ of $\overline{\kappa}_{m,d}$ seem to generate linearly independant elements $\mu \mapsto g_\mu(\vec{k})$ of $\mathcal{C}^0(\mathcal{P}, V)$. This could be compared to the discrete EIM (DEIM), where the EIM algorithm is applied on the indices list of some POD vectors, which are all kept for the approximation [11]. However, the main advantage of the EIM in our case is in the selection of relevant parameter values in a large set $\mathcal{P}_{\text{sample}}$, see Figure 4 showing the location of parameter values selected by EIM. In the numerical experiments of Section 5, we took $N = Q_{m,d}$ for this reason, and we can assess the convergence of the approximation with respect to the upper bounds of Corollary 4.4.

Nevertheless, we derived a rather general result of convergence in Proposition 4.1, which could be very usefull for other classes of models and problems. Notice also that with $\hat{Q}_{m,d}$ evaluation of the reference quantity $\mathcal{L}_\mu$, $\theta_{\mathcal{L}}^{s\hat{Q}_{m,d}}$ involves an approximation on a $s\hat{Q}_{m,d}$-dimensional subspace of $\mathcal{C}^0(\mathcal{P}, V)$, where s is the dimension of V.

5. NUMERICAL EXPERIMENTS

In this section, numerical comparisons between the presented algorithms and others taken from the literature are presented.

5.1. Inverse operators and solution to linear systems

Consider an open set $\Omega \subset \mathbb{R}^3$ meshed with tetrahedra, see Figure 1. This set represents a high pressure turbine blade featuring three cooling corridors; the intersection between these corridors and Ω is denoted $\partial\Omega_\mathcal{C}$. We consider the following archetypal heat problem:

$$\begin{cases} -\vec{\nabla} \cdot \vec{q} = \mu_2 u_\mu & \text{in } \Omega, \\ \vec{q} \cdot \vec{n} = -1 & \text{on } \partial\Omega_\mathcal{C}, \\ \vec{q} \cdot \vec{n} = 0 & \text{on } \partial\Omega \backslash \partial\Omega_\mathcal{C}, \end{cases} \tag{5.1}$$

where u_μ is the temperature, $\vec{q} = -\mu_1 \vec{\nabla} u_\mu$ is the heat flux density, and $\mu = (\mu_1, \mu_2) \in \mathcal{P} := (1, 4)^2$ is the parameter. In this problem, μ_1 is the heat conductivity, and $\mu_2 u_\mu$ is a volumic source term depending on the solution u_μ.
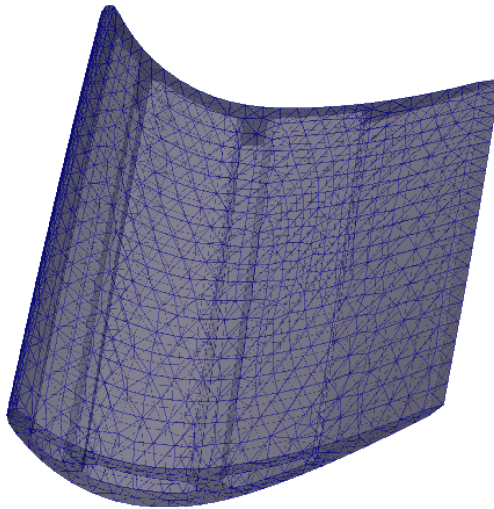

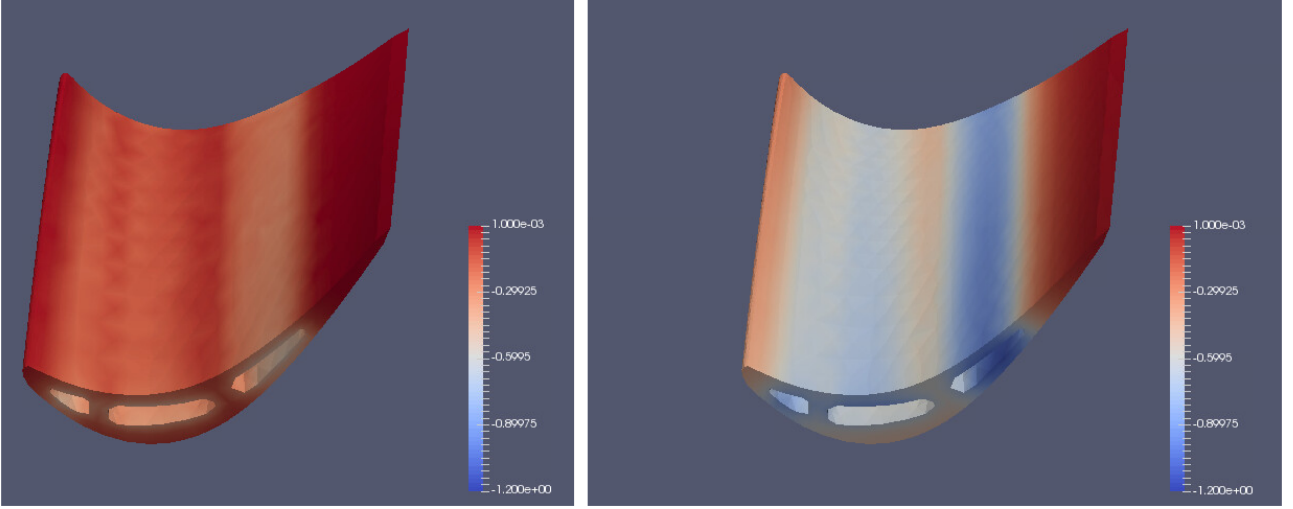FIGURE 1. Mesh of the high pressure turbine blade.

FIGURE 2. Solutions u_h to (5.2), for respective parameter values (1.82, 3.87) and (3.48, 1.21).

Denote $\mathcal{V}_h(\Omega)$ the space of P1-finite elements associated with the considered mesh of Ω, where h denotes the characteristics length of the tetrahedra constituting the mesh. The weak form of (5.1) can be approximated by

$$A_\mu U_\mu = b, \tag{5.2}$$

where $A_\mu = \mu_1 A_1 + \mu_2 A_2$, with $(A_1)_{i,j} = \int_\Omega \vec{\nabla}\Phi_i \cdot \vec{\nabla}\Phi_j$ and $(A_2)_{i,j} = \int_\Omega \Phi_i \Phi_j$, and $b_i = \int_{\partial\Omega_C} \Phi_i$; $\{\Phi_i\}_{1\le i\le \mathcal{N}}$ denoting the P1-finite elements basis, where $\mathcal{N} = 3,296$ in this example. The approximation $u_{\mu h} \in \mathcal{V}_h$ of the solution u_μ of (5.1) is obtained as $u_{\mu h} := \sum_{i=1}^{\mathcal{N}} U_{\mu i}\Phi_i$. Two solutions $u_{\mu h}$ at two different parameter values are shown in Figure 2.

In this section, we compare the approximation (3.14) with other methods for approximating parametrized solutions:

(1) (Minimisation in the Frobenius norm) Let $Y_{Q_{m,d}} = \mathrm{Span}\{A_{\mu_1}^{-1}, \cdots, A_{\mu_{Q_{m,d}}}^{-1}\}$, and define

$$P_{Q_{m,d}}(\mu) := \operatorname*{argmin}_{P\in Y_{Q_{m,d}}} \|I - PA(\mu)\|_F, \tag{5.3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. From [36], $P_{Q_{m,d}}(\mu) = \sum_{l=1}^{Q_{m,d}} \lambda_l(\mu)A_{\mu_l}^{-1}$, with $\lambda(\mu)$ the solution of

$$M(\mu)\lambda(\mu) = S(\mu), \tag{5.4}$$

with $M_{i,j}(\mu) = \mathrm{trace}(A_\mu^T A_{\mu_i}^{-T} A_{\mu_j}^{-1} A_\mu)$ and $S_i(\mu) = \mathrm{trace}(A_{\mu_i}^{-1} A_\mu)$. $P_{Q_{m,d}}(\mu)$ is then the best approximation of A_μ^{-1} expressed as a linear combination of inverses, in a chosen distance. However, the construction of $M(\mu)$ and $S(\mu)$ requires the online constructions of $A_{\mu_i}^{-1} A_\mu$, $1 \le i \le Q_{m,d}$, which is too computationally expensive – the goal of [36] is to propose computationaly effective approximations of (5.3). Here, we make use of (2.1) to write $M_{i,j}(\mu) = \sum_{l,m=1} \alpha_l(\mu)\alpha_m(\mu)\mathrm{trace}(A_l^T A_{\mu_i}^{-T} A_{\mu_j}^{-1} A_m)$ and $S_i(\mu) = \sum_{l=1}^{Q_{m,d}} \alpha_l(\mu)\mathrm{trace}(A_{\mu_i}^{-1} A_l)$, where the matrices $\left(\mathrm{trace}(A_l^T A_{\mu_i}^{-T} A_{\mu_j}^{-1} A_m)\right)_{i,j}$, $1 \le l,m \le Q_{m,d}$ and right-hand sides $\left(\mathrm{trace}(A_{\mu_i}^{-1} A_l)\right)_i$, $1 \le l \le Q_{m,d}$ can be precomputed – which is still much more computationally demanding than the proposed algorithm.

Finally, the approximation of (5.2) can be written $u_{\mu_h}(\mu) \approx \sum_{l=1}^{Q_{m,d}} \lambda_l(\mu) u_{\mu_h}(\mu_l)$, with $\lambda(\mu)$ the solution of (5.4). Notice that this method is intrusive since it requires to access the matrices A_μ, instead of only the solutions $u_{\mu_h}(\mu)$ for nonintrusive strategies.

(2) (Proper Orthogonal Decomposition (POD) [30]) First, we construct $M \in \mathbb{R}^{Q_{n,d} \times N}$ such that $M_{i,j} = u_{\mu_h}(\mu_i)_j$. Then, we compute the singular value decomposition (SVD) of M: $M = WSV$, with $S \in \mathbb{R}^{Q_{m,d} \times N}$ containing the singular values of M on its diagonal and zero elsewhere, and $W \in \mathbb{R}^{Q_{m,d} \times Q_{m,d}}$ and $V \in \mathbb{R}^{N \times N}$ unitary matrices. The $\hat{N}$ largest singular values are kept, following an accuracy criterion, and we denote $\hat{V} \in \mathbb{R}^{\hat{N} \times N}$ such that $\hat{V}_{i,j} = V_{i,j}$, $1 \le i \le \hat{N}$, $1 \le j \le N$, and $\{\hat{v}_i\}_{1 \le i \le \hat{N}}$, such that $(\hat{v}_i)_j = \hat{V}_{i,j}$, $1 \le i \le \hat{N}$, $1 \le j \le N$. Finally, the approximation of $u_{\mu_h}(\mu)$ is computed as $u_{\mu_h}(\mu) \approx \sum_{i=1}^{\hat{N}} \theta_i(\mu) \hat{v}_i$, where $\theta(\mu)$ solves $\hat{A}_\mu \theta(\mu) = \hat{b}$, with $\hat{A}_\mu = \alpha_1(\mu) \hat{V} A_1 \hat{V}^T + \alpha_2(\mu) \hat{V} A_2 \hat{V}^T$, and $\hat{b} = \hat{V} b$.

Notice that this method is intrusive since it requires to access the matrices A_μ, and that the result is not a linear combinations of solutions, but a linear combinations of singular vectors of M.

(3) (Meta-modelisation) This first step is the same as the previous item: the construction of a basis $\hat{v}_i$, $1 \le i \le \hat{N}$ using the SVD. To provide a fair comparison, we did not use the snapshots $u_{\mu_h}(\mu_i)$ for the μ_i selected by the EIM on $g(\vec{k}, \mu)$, but we used a Latin Hypercube Sampling of same size, computed with the MaxProj algorithm (see [27]) to select the μ_i, which is recommended when using statistical meta-models. The obtained set of parameter values is called the Design Of Experiment (DOE). Then, we compute the coefficients of the snapshots on the constructed basis: $\alpha_{i,j} = (u_{\mu_i h}, \hat{v}_j)$, $1 \le i \le Q_{m,d}$, $1 \le j \le \hat{N}$. Finally, nonintrusive meta-models are constructed on the coefficients $\alpha_{i,j}$, for which the predictions $\theta_i(\mu)$ at new parameter value μ are used in the obtained approximation as $u_{\mu_h}(\mu) \approx \sum_{i=1}^{\hat{N}} \theta_i(\mu) \hat{v}_i$. The considered statistical methods are taken from the machine learning community: (i) Gaussian processes, (ii) gradient boosting regression, (iii) random forests and (iv) Bayesian Ridge regression; and computed using the python package scikit-learn, see [26].

Figure 3 shows a comparison for the different aforementioned approximations. In abscissa is the size of the DOE for the statistical methods, and corresponds to $Q_{m,d}$, the number of parameter values selected by EIM for the proposed algorithm, the Minimization of Frobenius norm, and the POD. We pick 100 random values for the parameter and compute the solutions u_{μ_t}, $1 \le t \le 100$. For each size of the DOE, we compare the predicted solution using the described approximations, at each value μ_t, and compute the relative Euclidian-norms errors as $\epsilon_{\mu_t} = \dfrac{\|u_{\mu_t} - \hat{u}_{\mu_t}\|_2}{\|u_{\mu_t}\|_2}$, where $\hat{u}$ denotes here the considered approximation. Finally, the mean of relative errors, $\frac{1}{100} \sum_{t=1}^{100} \epsilon_{\mu_t}$, is represented on Figure 3.

We see that the intrusive methods that require to access the matrix A_μ perform much better than the nonintrusive methods in the example. Among the nonintrusive considered ones, our algorithm exhibits the best convergence rate.

Figure 4 shows the locations of the parameters selected by EIM and by the MaxProj DOE in the parameter domain $(1,4)^2$. We notice that the EIM selects more points close to the boundary of the domain while the MaxProj DOE is more uniform.

5.2. Logarithm of the determinant

In this section, we consider $A_\mu = 0.045 \left(1 - e^{-\mu_1^2}\right) A_1 + (1 - e^{-\mu_2}) A_2$, where A_1 and A_2 are the same as in Section 5.1, and $(\mu_1, \mu_2) \in \mathcal{P} := (1,4)^2$. Figure 5 shows $\log(\det(A_\mu))$, $\mu \in \mathcal{P}$: this illustrates the quantity which we look to approximate using nonintrusive interpolation formulae (even if in large-dimensional test cases, we cannot afford to plot it).

We compare the approximation (3.26) with the statistical nonintrusive approximation methods considered in Section 5.1: (i) Gaussian processes, (ii) gradient boosting regression, (iii) random forests and (iv) Bayesian Ridge regression, see Figure 6. Here again, we construct a DOE using the MaxProj algorithm for selecting the values of the parameter used with the statistical nonintrusive approximation methods. We pick 100 random
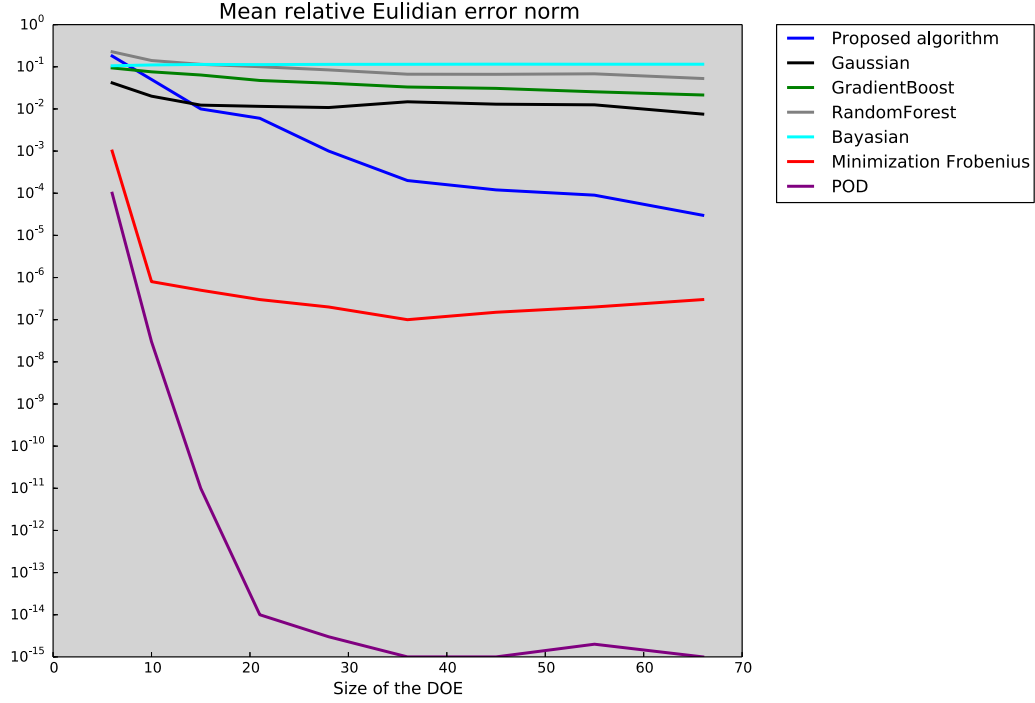
FIGURE 3. Mean relative Euclidian-norms errors on the solution to (5.2) using various interpolation methods, over a set of 100 randomly picked parameter values.

values for the parameter and compute the solutions u_{μ_t}, $1 \leq t \leq 100$. For each size of the DOE, we compute $\epsilon_{\mu_t} = \dfrac{\|\log(\det(A_{\mu_t})) - \hat{ld}_{\mu_t}\|_2}{\|\log(\det(A_{\mu_t}))\|_2}$, where $\hat{ld}$ denotes here the considered approximation. Finally, the mean of relative errors, $\frac{1}{100} \sum_{t=1}^{100} \epsilon_{\mu_t}$, is represented on Figure 6. We notice that the proposed algorithm exhibits the best convergence rate.

5.3. Experiments in high parameter dimension cases

5.3.1. A thermal problem in parameter dimension 10

We consider the same geometry as in Section 5.1, see Figure 1, and the following problem:

$$\begin{cases} c_p \dfrac{\partial u_\mu}{\partial t} - \vec{\nabla} \cdot \left(\eta \vec{\nabla} u_\mu \right) = 0 & \text{in } \Omega, \\ \eta \vec{\nabla} u_\mu = 1000 \text{ W.m}^{-2} & \text{on } \partial\Omega_{\mathcal{C}}, \\ \vec{\nabla} u_\mu = 0 & \text{on } \partial\Omega \backslash \partial\Omega_{\mathcal{C}}, \\ u_\mu = 0 \,^{\circ}C & \text{at } t = 0, \end{cases} \qquad (5.5)$$

where c_p denotes here the heat capacity multiplied by the density, η is the thermal conductivity, and u_μ is the unknown temperature field. We choose $\eta = 370$ W.m^{-1}.K^{-1}, and c_p contains the parameter dependence as
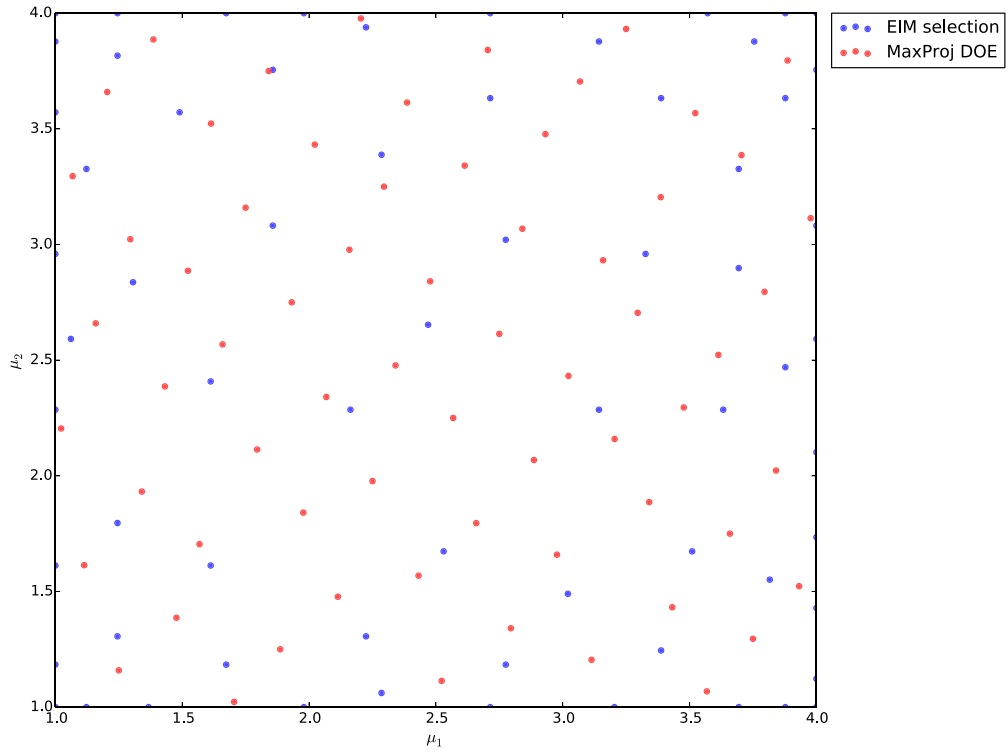
FIGURE 4. Locations of the parameters selected by EIM and by the MaxProj DOE in the parameter domain $(1, 4)^2$, for the largest DOE size considered in Figure 3 (66 points).
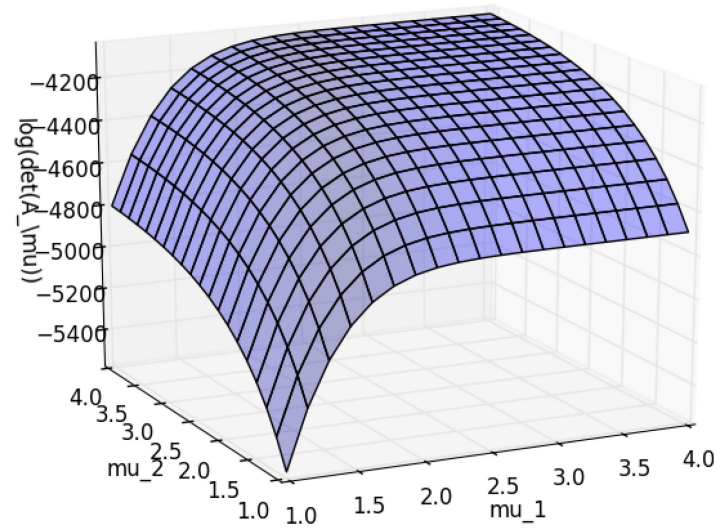

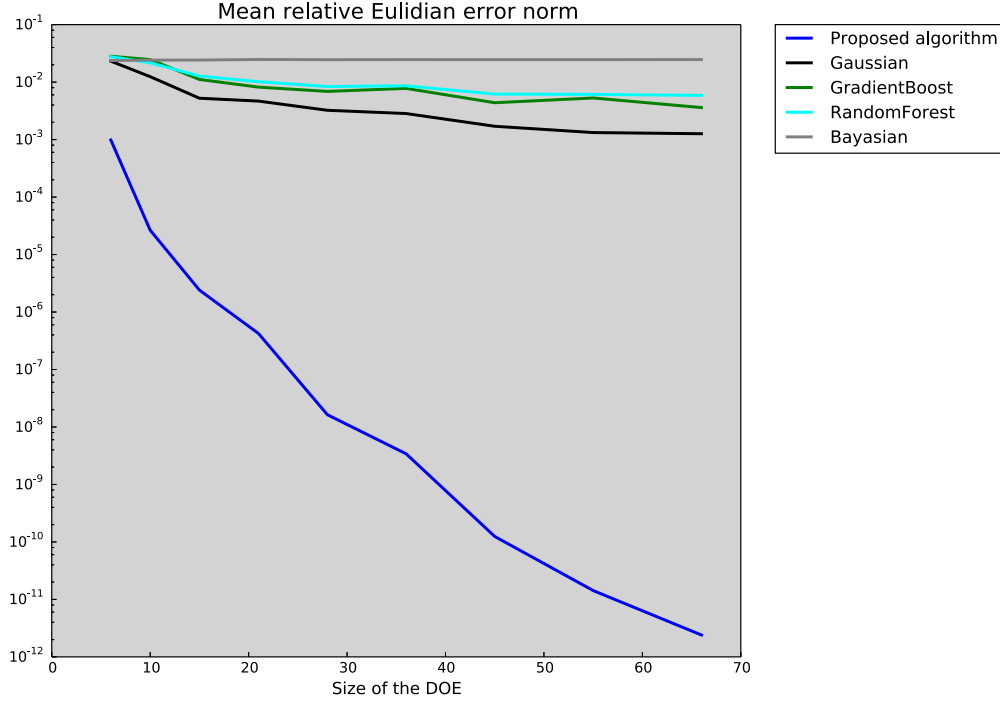FIGURE 5. Representation of $\log(\det(A_\mu))$, with $\mu \in \mathcal{P} := (1, 4)^2$.

FIGURE 6. Mean relative errors on the computation of $\log(\det(A_\mu))$ using the herein proposed algorithm and various nonintrusive machine learning regression methods.

follows (in $\text{J.m}^{-3}.\text{K}^{-1}$):

– experiment 1: $c_p(\mu, x) = 10 + \mu_1 \cos(0.2x) + \mu_2 \cos(0.25y) + \mu_3 \cos(0.3z) + \mu_4 \cos(0.2(x+y)) + \mu_5 \cos(0.25(x+z)) + \mu_6 \cos(0.3(y+z)) + \mu_7 \frac{x}{x_{\max}} + \mu_8 \frac{y}{y_{\max}} + \mu_9 \frac{z}{z_{\max}} + \mu_{10} \cos(0.1(x+y+z))$,
  $\mu \in \mathcal{P} = (0.1, 0.15)^{10}, (0.1, 0.2)^{10}, (0.1, 0.3)^{10}, (0.1, 0.6)^{10}$ or $(0.1, 1.1)^{10}$,

– experiment 2: $c_p(\mu, x) = 10 + (1 - e^{-\mu_1}) \cos(0.2x) + (1 - e^{-\mu_2}) \cos(0.25y) + (1 - e^{-\mu_3}) \cos(0.3z) + (1 - e^{-\mu_4}) \cos(0.2(x+y)) + (1 - e^{-\mu_5}) \cos(0.25(x+z)) + (1 - e^{-\mu_6}) \cos(0.3(y+z)) + (1 - e^{-\mu_7}) \frac{x}{x_{\max}} + (1 - e^{-\mu_8}) \frac{y}{y_{\max}} + (1 - e^{-\mu_9}) \frac{z}{z_{\max}} + (1 - e^{-\mu_{10}}) \cos(0.1(x+y+z))$,
  $\mu \in \mathcal{P} = (2, 2.05)^{10}, (2, 2.1)^{10}, (2, 2.2)^{10}, (2, 2.5)^{10}$ or $(2, 3)^{10}$,

where $(x, y, z) \in \Omega$ is the space variable, and $x_{\max}$, $y_{\max}$ and $z_{\max}$ denote the maximum of the components of the space variable over Ω. Examples of heat capacities taken from experiment 1 are represented in Figure 7.

We are interested in the solution of (5.5) at $t = 100\,\text{s}$. Using a backward Euler time-discretization, the weak form reads: find $u_\mu \in H_0^1(\Omega)$ such that for all $v \in H_0^1(\Omega)$,

$$\int_\Omega c_p \left(\frac{u_\mu(\vec{x}, t = 100) - u_\mu(\vec{x}, t = 0)}{\Delta t} \right) v(\vec{x}) + \int_\Omega \eta \vec{\nabla} u_\mu(\vec{x}, t = 100) \cdot \vec{\nabla} v(\vec{x}) = \int_{\partial \Omega_\mathcal{C}} \eta \vec{\nabla} u_\mu(\vec{x}, t = 100) \cdot \vec{n}, \quad (5.6)$$

where $\vec{n}$ is the exterior normal on $\partial \Omega_\mathcal{C}$, and $H_0^1(\Omega) = \{v \in L^2(\Omega) \text{ such that } \vec{\nabla} v \in L^2(\Omega) \text{ and } v|_{\partial \Omega \setminus \partial \Omega_\mathcal{C}} = 0\}$. A finite element approximation is obtained as $A_\mu U_\mu = b$, where

$$
\begin{aligned}
(A_\mu)_{i,j} &= \int_{\vec{x} \in \Omega} \frac{c_p(\vec{x}, \mu)}{100} \Phi_i(\vec{x}) \Phi_j(\vec{x}) dx + 370 \int_{\vec{x} \in \Omega} \vec{\nabla} \Phi_i(\vec{x}) \cdot \vec{\nabla} \Phi_j(\vec{x}) d\vec{x}, \qquad 1 \le i, j \le \mathcal{N}, \\
b_j &= 1000 \int_{\vec{x} \in \partial \Omega_\mathcal{C}} \Phi_j(\vec{x}) d\vec{x}, \qquad\qquad\qquad\qquad\qquad\qquad 1 \le j \le \mathcal{N},
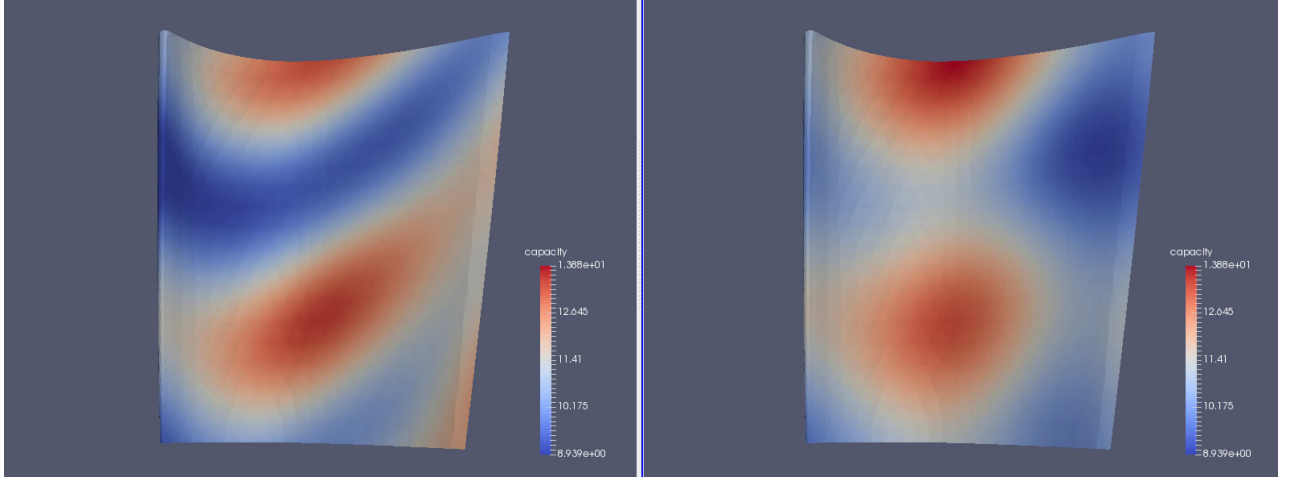\end{aligned}
\tag{5.7}
$$

FIGURE 7. Example of heat capacities taken from experiment 1.

where we recall that Φ_i denoted the P1-finite element basis. An approximation of (5.5) is obtained as $u_\mu(\vec{x}, t = 100) = \sum_{i=1}^{\mathcal{N}} U_{\mu_i} \Phi_i(\vec{x})$. For experiment 1, the affine decomposition (2.1) is obtained as $A_\mu = \sum_{l=1}^{11} \alpha_l(\mu) A_l$ with

$$(A_1)_{i,j} = \int_{\vec{x} \in \Omega} \frac{1}{10} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} + 370 \int_{\vec{x} \in \Omega} \vec{\nabla}\Phi_i(\vec{x}) \cdot \vec{\nabla}\Phi_j(\vec{x})dx \qquad (A_2)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.2x)}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x}$$

$$(A_3)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.25y)}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_4)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.3z)}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_5)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.2(x+y))}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x}$$

$$(A_6)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.25(x+z))}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_7)_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.3(y+z))}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_8)_{i,j} = \int_{\vec{x} \in \Omega} \frac{x}{100 x_{\max}} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x}$$

$$(A_9)_{i,j} = \int_{\vec{x} \in \Omega} \frac{y}{100 y_{\max}} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_{10})_{i,j} = \int_{\vec{x} \in \Omega} \frac{z}{100 z_{\max}} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x} \qquad (A_{11})_{i,j} = \int_{\vec{x} \in \Omega} \frac{\cos(0.1(x+y+z))}{100} \Phi_i(\vec{x})\Phi_j(\vec{x})d\vec{x}$$

and $\alpha_1(\mu) = 1, \alpha_{l+1}(\mu) = \mu_{(l)}, 1 \leq l \leq 10$ (notice that $\mu_{(l)}$ denotes the component l for the continuous parameter $\mu \in \mathbb{R}^{10}$). For experiment 2, the matrices A_l, $1 \leq l \leq 11$, are the same and $\alpha_1(\mu) = 1, \alpha_{l+1}(\mu) = (1 - e^{-\mu_{(l)}})$, $1 \leq l \leq 10$.

Figure 8 shows the relative errors between the proposed algorithm and Gaussian processes, as detailed in Section 5.1, in L^2- and L^∞-norms. The DOE for the Gaussian processes in the parameter spaces is obtained using MaxProj as well, and we do not provide a comparison with the other statistical methods considered in the previous sections since they exhibited worse results. We notice that the proposed algorithm provides accurate results. Then, smaller parameter discrepancies lead to more accurate results: the reducibility of the problem is better, and ρ in (3.3) should be smaller leading to smaller bounds $C_1(m)$ in (4.8) (at each m). Moreover, at fixed parameter discrepancy (hence fixed ρ), the errors decrease as $Q_{m,d}$ increases (hence as m increases): the EIM computes exactly (*i.e.* with no approximation errors, due to the interpolation property 3.4) more elements of the series defined by the iteration scheme (3.4), and the $A_{\mu_l}^{-1}$ in (3.13) are closer to the $X_m g_{\mu_l}$ in (3.12). In (4.8), this corresponds to the convergence of $C_1(m)$ to 0 with respect to m.

5.3.2. A mechanical problem in parameter dimension 14

Consider a cube Ω meshed with linear hexahedra, with all displacement boundary conditions fixed on one face (denoted Γ_D) and a prescribed stress on the opposite face (denoted Γ_N), the other faces are free. The domain contains 6 fibers $\Omega_1, \cdots, \Omega_6$, see Figure 9. We define $\Omega_0 := \Omega \backslash (\cup_{i=1}^6 \Omega_i)$. We consider the following linear elasticity problem: find $u \in H_0^1(\Omega)^3$ such that $\forall v \in H_0^1(\Omega)^3$

$$\int_\Omega \frac{\eta_1}{2} \left(\nabla u_\mu + {}^t\nabla u_\mu\right) \cdot \left(\nabla v + {}^t\nabla v\right) + \int_\Omega \eta_2 \left(\nabla \cdot u_\mu\right) \left(\nabla \cdot v\right) = \int_{\Gamma_N} (t \cdot n)v, \tag{5.8}$$

where, $H_0^1(\Omega)^3 = \{w \in L^2(\Omega)^3 \text{ such that } \nabla w \in L^2(\Omega)^{3 \times 3} \text{ and } w|_{\Gamma_D} = 0\}$, η_1 and η_2 are respectively Lamé's first and second parameters, $t = t_0 n$ (with n the outward unit normal and $t_0 = -100 \ N.m^{-2}$) is the prescribed
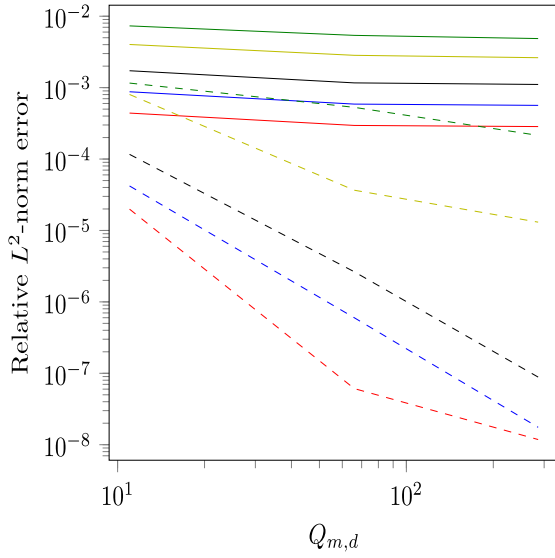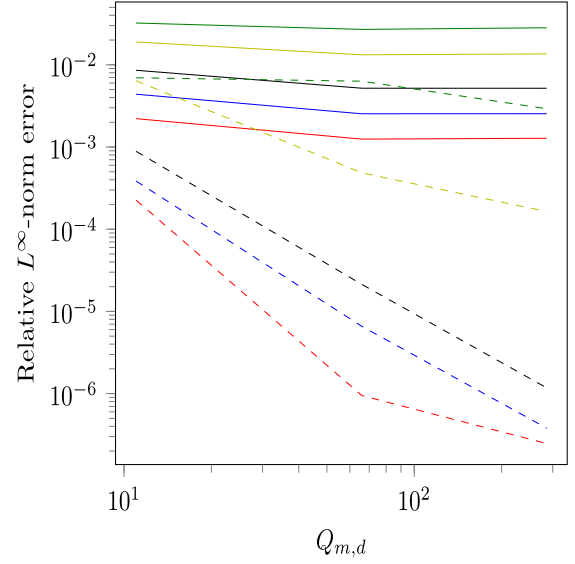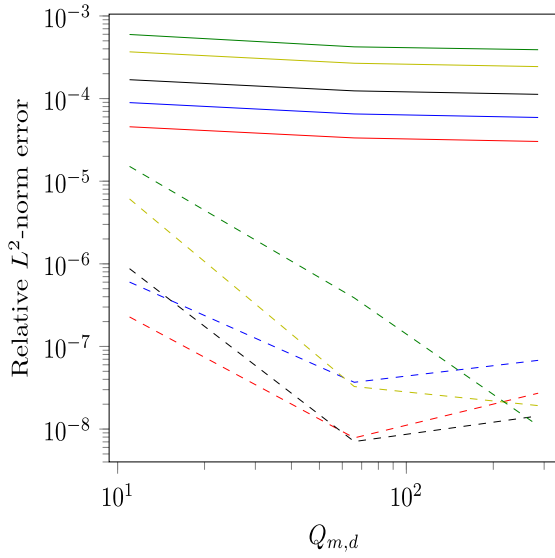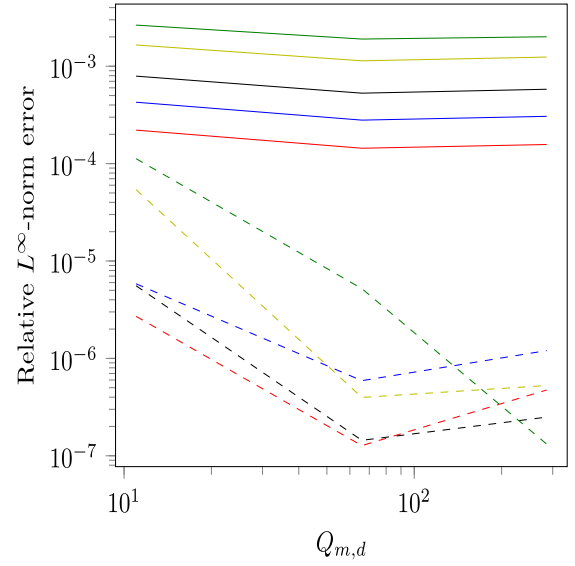
(A) Relative L^2-norm error.

(B) Relative L^∞-norm error.

(C) Relative L^2-norm error.

(D) Relative L^∞-norm error.

FIGURE 8. [(A, B): experiment 1 ; (C, D): experiment 2] Comparison between the proposed algorithm and Gaussian processes in a parameter dimension 10 test case: —— Gaussian 0.05, —— Gaussian 0.1, —— Gaussian 0.2, —— Gaussian 0.5, —— Gaussian 1, --- Proposed algorithm 0.05, --- Proposed algorithm 0.1, --- Proposed algorithm 0.2, --- Proposed algorithm 0.5, --- Proposed algorithm 1, the number indicates the parameter discrepancy.
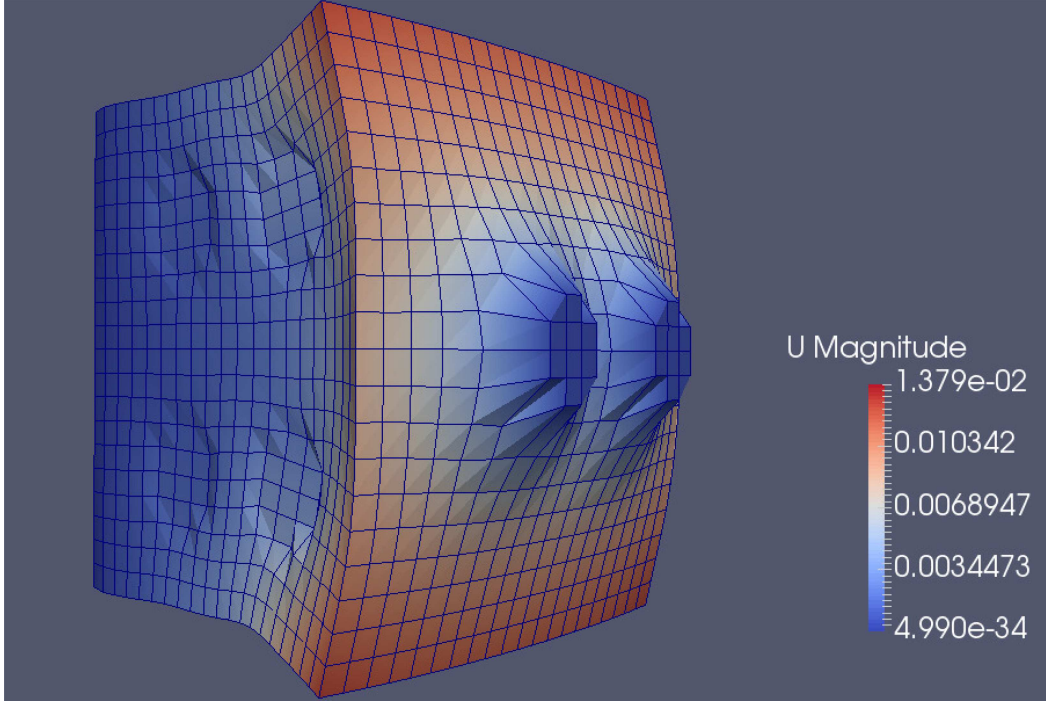
FIGURE 9. Finite element approximation of the solution of equation (5.8), where the mesh is deformed proportionally to the solution.

traction vector on Γ_N, and u_μ is the unknown displacement. See Figure 9 for a representation of a finite element approximation of the solution of equation (5.8). We denote $\eta_{1,k}$ and $\eta_{2,k}$ respectively Lamé's first and second parameters of the subdomains Ω_k, $0 \leq k \leq 6$. We choose as parameter $\mu = (\eta_{1,0}, \eta_{2,0}, \eta_{1,1}, \eta_{2,1}, \cdots, \eta_{1,6}, \eta_{2,6})$, and the affine decomposition (2.1) is obtained as $A_\mu = \sum_{l=1}^{14} \alpha_l(\mu) A_l$ with $(A_{2k})_{i,j} = \int_{\Omega_k} \frac{1}{2} (\nabla \Phi_i + {}^t\nabla \Phi_i) \cdot (\nabla \Phi_j + {}^t\nabla \Phi_j)$ and $(A_{2k+1})_{i,j} = \int_{\Omega_k} (\nabla \cdot \Phi_i)(\nabla \cdot \Phi_j)$, $0 \leq k \leq 6$, $1 \leq i, j \leq \mathcal{N}$ (where $(\Phi_i)_{1 \leq i \leq \mathcal{N}}$ is the basis of a finite element space approximating $H_0^1(\Omega)^3$, with $\mathcal{N} = 27,783$), and $\alpha_{2k} = \eta_{1,k}$, $\alpha_{2k+1} = \eta_{2,k}$, $0 \leq k \leq 6$. The parameter set is defined as follows: the reference Poisson coefficient is 0.3 in the whole cube, and the Young modulus for the fibers is 2×10^9, and 2×10^6 in the rest of the domain. From these values, we compute the reference Lamé's coefficients $(\eta_{1,k}, \eta_{2,k}) = (1.15 \times 10^9, 7.7 \times 10^8)$ for the fibers (namely $1 \leq k \leq 6$), and $(\eta_{1,0}, \eta_{2,0}) = (1.15 \times 10^6, 7.7 \times 10^5)$ for the rest of the domain. Three parameter sets are considered, constituted of the intervals centered at the reference parameter values previously defined, with length respectively 1%, 5% and 10% of the corresponding reference value.

In Figure 10 are represented the relative errors between the proposed algorithm and Gaussian processes, as done in Section 5.3.1, in L^2- and L^∞-norms. The same conclusions as Section 5.3.1 can be drawn.

CONCLUSION

In this work, we propose an algorithm to approximate, in a nonintrusive fashion, the limits of parametrized series of linear operators with respect to a functional g, based on the EIM approximation of g. We derive upper bounds of the error made by the obtained algorithm. With a strong enough convergence of the considered series, we prove the convergence of our algorithm. This assumption is verified by the two application considered in this work: the inverse and the logarithm of the determinant of a family of parametrized matrices. The numerical
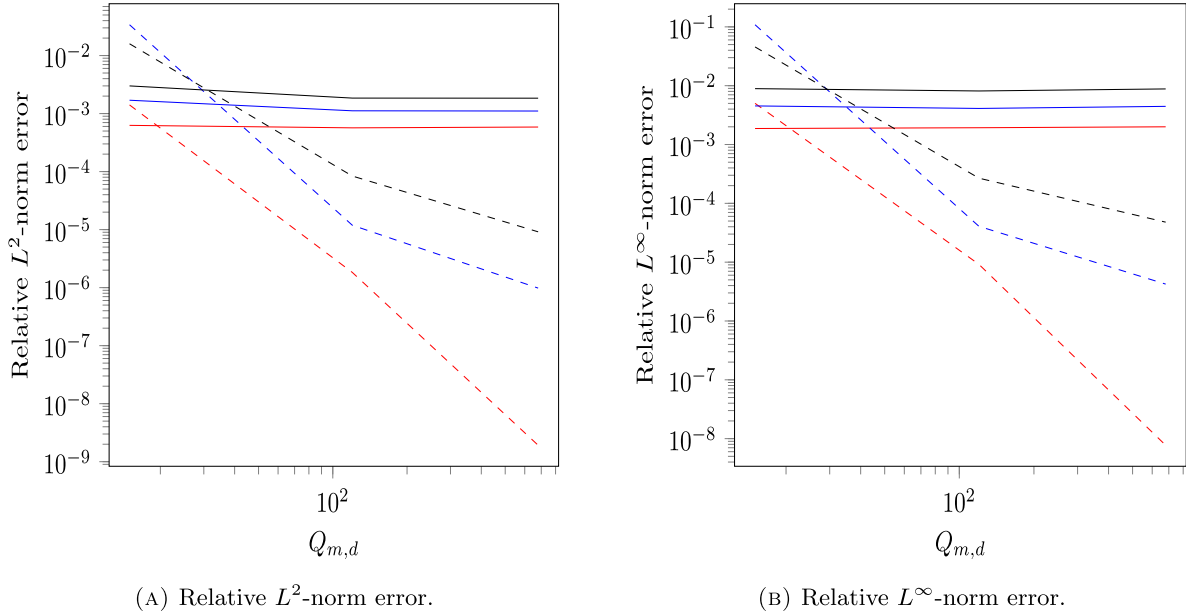
(A) Relative L^2-norm error.

(B) Relative L^∞-norm error.

FIGURE 10. Comparison between the proposed algorithm and Gaussian processes in a parameter dimension 14 test case: —— Gaussian 1%, —— Gaussian 5%, —— Gaussian 10%, - - - Proposed algorithm 1%, - - - Proposed algorithm 5%, - - - Proposed algorithm 10%, the % indicates the parameter discrepancy.

simulations illustrate that, in the considered test cases, our algorithm performs well compared to classical nonintruive approximations taken from the machine learning community.

REFERENCES

[1] C.V. Ananth and D.G. Kleinbaum, Regression models for ordinal responses: a review of methods and applications. *Int. J. Epidemiol.* **26** (1997) 1323–1333.

[2] T.W. Anderson and I. Olkin, Maximum-likelihood estimation of the parameters of a multivariate normal distribution. *Linear Algebra Appl.* **70** (1985) 147–171.

[3] M. Barrault, Y. Maday, N.-C. Nguyen and A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *C.R. Math.* **339** (2004) 667–672.

[4] M. Bergmann, *Optimisation aérodynamique par réduction de modèle POD et contrôle optimal: application au sillage laminaire d'un cylindre circulaire.* Ph.D. thesis, Vandoeuvre-les-Nancy, INPL (2004).

[5] C. Boutsidis, P. Drineas, P. Kambadur, E.-M. Kontopoulou and A. Zouzias, A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix. *Linear Algebra Appl.* **533** (2017) 95–117.

[6] A. Buffa, Y. Maday, A.T. Patera, C.Prud'homme and G. Turinici, *A priori* convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: M2AN* **46** (2012) 595–603.

[7] F. Casenave, A. Ern and T. Lelièvre, A nonintrusive reduced basis method applied to aeroacoustic simulations. *Adv. Comput. Math.* **41** (2015) 961–986.

[8] F. Casenave, A. Ern and T. Lelièvre, Variants of the Empirical Interpolation Method: Symmetric formulation, choice of norms and rectangular extension. *Appl. Math. Lett.* **56** (2016) 23–28.

[9] R. Chakir, P. Joly, Y. Maday and P. Parnaudeau, A Non intrusive reduced basis method: application to computational fluid dynamics. In: 2nd ECCOMAS Young Investigators Conference (YIC 2013), Bordeaux, France (2013).

[10] A. Chatterjee, An introduction to the proper orthogonal decomposition. *Curr. Sci.* **78** (2000) 808–817.

[11] S. Chaturantabut and D.C. Sorensen, Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* **32** (2010) 2737–2764.

[12] M. Chevreuil and A. Nouy, Model order reduction based on proper generalized decomposition for the propagation of uncertainties in structural dynamics. *Int. J. Numer. Methods Eng.* **89** (2012) 241–268.

[13] F. Chinesta, A. Ammar, A. Leygue and R. Keunings, An overview of the Proper Generalized Decomposition with applications in computational rheology. XVIth International Workshop on Numerical Methods for Non-Newtonian Flows. *J. Non-Newtonian Fluid Mech.* **166** (2011) 578–592.

[14] F. Chinesta, R. Keunings and A. Leygue, The Proper Generalized Decomposition for Advanced Numerical Simulations: A Primer. SpringerBriefs in Applied Sciences and Technology. Springer International Publishing, New York (2013).

[15] F. Chinesta, P. Ladeveze and C. Elias, A short review on model order reduction based on proper generalized decomposition. *Arch. Comput. Methods Eng.* **18** (2011) 395–404.

[16] A. Dasgupta, Y.V. Sun, I.R. König, J.E. Bailey-Wilson and J.D. Malley, Brief review of regression-based and machine learning methods in genetic epidemiology: the genetic analysis workshop 17 experience. *Genet. Epidemiol.* **35** (2011) S5–S11.

[17] D.J. Knezevic, N.C. Nguyen and A.T. Patera, Reduced basis approximation and a posteriori error estimation for the parametrized unsteady Boussinesq equations. *Math. Models Methods Appl. Sci.* **21** (2011) 1415–1442.

[18] G. Luo, A review of automatic selection methods for machine learning algorithms and hyper-parameter values. *Network Model. Anal. Health Inf. Bioinf.* **5** (2016) 18.

[19] L. Machiels, Y. Maday, I.B. Oliveira, A.T. Patera and D.V. Rovas, Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *C. R. Acad. Sci. Paris Ser. I* **331** (2005) 153–158.

[20] L. Machiels, Y. Maday, A.T. Patera, C. Prud'homme, D.V. Rovas, G. Turinici and K. Veroy, Reliable real-time solution of parametrized partial differential equations: reduced-basis output bound methods. *CJ Fluids Eng.* **124** (2002) 70–80.

[21] Y. Maday, O. Mula and G. Turinici, Convergence analysis of the Generalized Empirical Interpolation Method. *SIAM J. Numer. Anal.* **54** (2016) 1713–1731.

[22] Y. Maday, N.-C. Nguyen, A.T. Patera and S.H. Pau, A general multipurpose interpolation procedure: the magic points. *Commun. Pure Appl, Anal.* **8** (2009) 383–404.

[23] R.J. Martin, Approximations to the determinant term in Gaussian maximum likelihood estimation of some spatial models. *Commun. Stat. Theory Methods* **22** (1992) 189–205.

[24] P. Meer, D. Mintz, A. Rosenfeld and D.Y. Kim, Robust regression methods for computer vision: a review. *Int. J. Comput. Vision* **6** (1991) 59–70.

[25] A.T. Patera, C. Prud'homme, D.V. Rovas and K. Veroy, A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, Florida, USA (2003).

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, Scikit-learn: machine learning in python. *J. Mach. Learn. Res.* **12** (2011) 2825–2830.

[27] V. Roshan Joseph, E. Gul and S. Ba, Maximum projection designs for computer experiments. *Biometrika* **102** (2015) 371–380.

[28] S. Sen, Reduced-basis approximation and a posteriori error estimation for many-parameter heat conduction problems. *Numer. Heat Transfer Part B: Fundam.* **54** (2008) 369–389.

[29] S. Sen, K. Veroy, D.B.P. Huynh, S. Deparis, N.C. Nguyen and A.T. Patera, "Natural norm" a posteriori error estimators for reduced basis approximations. *J. Comput. Phys.* **217** (2006) 37–62.

[30] L. Sirovich, Turbulence and the dynamics of coherent structures, parts I, II and III. *Q. Appl. Math.* **XLV** (1987) 561–590.

[31] I. Uysal and H. A. Güvenir, An overview of regression techniques for knowledge discovery. *Knowl. Eng. Rev.* **14** (1999) 319–340.

[32] K. Veroy and A.T. Patera, Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations: rigorous reduced-basis a posteriori error bounds. *Int. J. Numer. Methods Fluids* **47** (2005) 773–788.

[33] K. Veroy, C. Prud'homme and A.T. Patera, Reduced-basis approximation of the viscous Burgers equation: rigorous a posteriori error bounds. *C.R. Math.* **337** (2003) 619–624.

[34] D. Xiao, Z. Lin, F. Fang, C.C. Pain, I.M. Navon, P. Salinas and A. Muggeridge, Non-intrusive reduced-order modeling for multiphase porous media flows using Smolyak sparse grids. *Int. J. Numer. Methods Fluids* **83** (2017) 205–219.

[35] M. Yano, A space-time Petrov–Galerkin certified reduced basis method: Application to the Boussinesq equations. *SIAM J. Sci. Comput.* **36** (2014) A232–A266.

[36] O. Zahm and A. Nouy, Interpolation of inverse operators for preconditioning parameter-dependent equations. *SIAM J. Sci. Comput.* **38** (2016) A1044–A1074.

[37] Y. Zhang, W.E. Leithead, D.J. Leith and L. Walshe, Log-det approximation based on uniformly distributed seeds and its application to Gaussian process regression. *J. Comput. Appl. Math.* **220** (2008) 198–214.