SIMULTANEOUS REDUCED BASIS APPROXIMATION OF PARAMETERIZED ELLIPTIC EIGENVALUE PROBLEMS *

THOMAS HORGER[1], BARBARA WOHLMUTH[1] AND THOMAS DICKOPF[1]

Abstract. The focus is on a model reduction framework for parameterized elliptic eigenvalue problems by a reduced basis method. In contrast to the standard single output case, one is interested in approximating several outputs simultaneously, namely a certain number of the smallest eigenvalues. For a fast and reliable evaluation of these input-output relations, we analyze *a posteriori* error estimators for eigenvalues. Moreover, we present different greedy strategies and study systematically their performance. Special attention needs to be paid to multiple eigenvalues whose appearance is parameter-dependent. Our methods are of particular interest for applications in vibro-acoustics.

1. INTRODUCTION

For the fast and reliable evaluation of input-output relations for parameterized partial differential equations (μPDEs), reduced basis methods have been developed over the last decade (see, *e.g.*, [34, 37] or [33], Chap. 19 for comprehensive reviews), with the first reduced basis problem being investigated in the 1980's [28]. The methodology has been applied successfully to many different problem classes both in the real-time and the many-query context. These problem classes include among others finite element discretizations of elliptic equations [37], parabolic equations [15, 35, 39] and hyperbolic equations [7, 16]. Furthermore the reduced basis method has been extended to Stokes problems [20, 23, 36, 38] as well as to variational inequalities [13, 17] with a time-space formulation of the problem and corresponding analysis. It was also applied to Stochastic Processes in [11, 40] and to a finite volume scheme of a parameterized and highly nonlinear convection-diffusion problem with discontinuous solutions in [9].

A posteriori error estimators w. r. t. parameter variations generally facilitate the construction of reduced basis spaces by greedy algorithms as well as the certification of the outputs of the reduced models. Different greedy methods for reduced basis and error estimators have been introduced in [26, 27, 41], also a greedy method for eigenvalues is introduced in [6], and the convergence of greedy methods has been analyzed in [4, 5, 8]. Another

[1] Lehrstuhl für Numerische Mathematik, Technische Universität München, Boltzmannstraße 3, 85748 Garching, Germany. horger@ma.tum.de

way to construct a reduced space is the proper orthogonal decomposition (POD) method, as discussed, *e.g.*, in [21, 33].

The problem class of parameterized elliptic eigenvalue problems (μEVPs) is highly important but up to now only marginally investigated in the context of reduced basis methods. The first approach [24] from the year 2000, which is based on [25] among others, is restricted to the special case of an estimator for the first eigenvalue. In the following publications [30–32], the method from [24] is developed further to include several eigenvalues.

However, both the analysis and the algorithms do not cover the case of multiple eigenvalues. Quite often, the "vectorial approach", *i.e.*, the treatment of the eigenvectors $(u_i(\mu))_{1 \leq i \leq K}$ as an (FE dimension $\cdot K$)-dimensional object and building the approximation space accordingly (*cf.* [31], Sect. 2.3.5), results in poor accuracy. This is due to the fact that the possible savings from reduced problems of smaller size seem marginal if achievable at all. In addition high-frequency information can and should be exploited for the approximation of low-frequency information, an effect that is expected to become more and more important with increasing number of desired eigenvalues. In [45], an elastic buckling problem is studied. While the model reduction is carried out solely/primarily for a linear problem, the eigenvalue problem appears only in a second step.

Here, a new RB space is built from the eigenfunctions associated with the smallest eigenvalues at the previously identified parameters. A non-rigorous *a posteriori* bound is then computed by comparison with a reduced space approximation of double size; *cf.* also [43]. Furthermore a component based RB method is studied for eigenvalue problems in [42].

Very recently an RB method for the approximation of single eigenvalues in the context of parameterized elliptic eigenvalue problems has been investigated in [12]. The authors derive a bound for the error in the first eigenvalue which is assumed to be single.

The aim of this paper is to develop a model reduction framework for elliptic μEVPs. The application scenario we target is the vibro-acoustics of cross-laminated timber structures. Here, a parameter-dependent eigenvalue problem in linear elasticity, where the input parameters are the material properties of different structural components, is to be solved many times during a design/optimization phase. Since the main part of a vibro-acoustical analysis is the modal analysis, which not only takes the first eigenvalue into account, but all eigenvalues under a certain frequency depending on the problem under investigation, the outputs of interest are the K smallest eigenvalues with corresponding eigenfunctions. A characteristic feature of the considered μEVPs is the appearance of multiple eigenvalues. In particular, the multiplicities depend on the parameters.

A particular challenge of the considered μEVPs is the rather large number of outputs of interest K, which in our exemplary case ranges from two to twenty. We are interested in approximating these smallest eigenvalues "simultaneously" in the sense that a single reduced space is constructed for the variational approximation of the eigenvalue problem and that the individual *a posteriori* error estimators for the eigenvalues use the online components provided offline. This allows us to generate an efficient and accurate simultaneous reduced basis approximation. The large number of outputs of interest K justifies an increased computational effort by an increased dimension N as compared to the standard single output case. In particular, any decent, *i.e.*, sufficiently accurate, approximation needs $N \gg K$. This approach constitutes a significant difference to the one taken in [12] as our goal is a reduced basis approximation not only of one eigenvalue, but of a series of eigenvalues, including eigenvalues with multiplicity greater than one. The parameter-dependence of the multiplicity of the eigenvalues constitutes a major challenge in this context and is included into both our analysis and our algorithms.

Furthermore our experiments show that, in a greedy algorithm, it is usually not optimal to include the first K eigenfunctions for a particular parameter, neither is it advisable to choose the same number of eigenfunctions for different eigenvalues. This may be attributed to the fact that the smoothness of the input-output relation can vary strongly with the different outputs of interest, *i.e.*, the eigenvalues. We rather suggest to choose maximizing parameters for K different error estimators, as described in Section 4.1. The reduced approximation should be of comparable quality for a broad range of frequencies, although in structural acoustics the accuracy requirements might decrease with increasing frequency. Note that, for the application scenario at hand, the number of desired eigenpairs is typically in the order of ten for simple components and even larger for geometrically more complex structures.

The main contributions of this paper are the analysis of an asymptotically reliable error estimator including the case of multiple eigenvalues and a series of algorithmic advancements. Our numerical results demonstrate that tailored greedy strategies yield very efficient reduced basis spaces for the simultaneous approximation of many eigenvalues for the considered problem class.

The rest of the paper is structured as follows: In Section 2, we describe the problem setting and introduce the reduced basis method for μEVPs. Section 3 is devoted to the *a posteriori* error analysis w.r.t. parameter variations. We also discuss how to evaluate the derived error estimators computationally. In Section 4, several greedy algorithms are presented. We demonstrate the effectivity of our algorithms by numerical examples with the application to linear elasticity in Section 5.

2. Problem setting

2.1. Parameterized eigenvalue problems in computational mechanics

Let the computational domain $\Omega \subset \mathbb{R}^d$, with $d = 2, 3$, be bounded and polygonal. As an elliptic eigenvalue model problem, we consider the linear elasticity case. But all our results also hold true for more general elliptic systems. Then, the eigenvalue problem in linear elasticity is given by

$$-\operatorname{div} \sigma = \lambda \rho u \quad \text{in } \Omega \tag{2.1}$$

with boundary conditions prescribed as Dirichlet conditions on a closed non-trivial subset Γ_D of $\partial\Omega$ and homogeneous Neumann conditions on $\partial\Omega \setminus \Gamma_D$. In addition, the linearized stress and strain tensors are defined as

$$\sigma(u) = \mathbb{C}(\mu)\epsilon(u) \qquad \text{and} \qquad \epsilon(u) = \frac{1}{2}(\nabla u + \nabla u^T),$$

respectively. We set the density ρ to 1 for simplicity. Furthermore, the set of admissible parameters is denoted by $\mathcal{P} \subset \mathbb{R}^P$ and $\mu \in \mathcal{P}$ stands for a vector of parameters. Then, $\mathbb{C}(\mu)$ denotes the parameter-dependent Hooke's tensor, which we assume to be uniformly positive definite. To this end, let Ω be decomposed into non-overlapping subdomains such that $\overline{\Omega} = \bigcup_s \overline{\Omega}_s$. We assume that the material parameters are piecewise constant w.r.t. this decomposition. In the isotropic case, the parameters may be chosen as Young's modulus E and Poisson's ratio ν such that P equals two times the number of structural components (*i.e.*, subdomains). More precisely, we set $\mu_{2s-1} = E|_{\Omega_s}$ and $\mu_{2s} = \nu|_{\Omega_s}$ in this case. The anisotropic case is treated analogously.

Let the bilinear forms $a(\cdot, \cdot; \mu) : (H^1(\Omega))^d \times (H^1(\Omega))^d \to \mathbb{R}$ and $m(\cdot, \cdot) : (L^2(\Omega))^d \times (L^2(\Omega))^d \to \mathbb{R}$ be given by

$$(u, v) \mapsto a(u, v; \mu) := \int_\Omega \mathbb{C}(\mu)\epsilon(u) : \epsilon(v) \, \mathrm{d}x$$

and

$$(u, v) \mapsto m(u, v) := (u, v)_{L^2(\Omega)} := \int_\Omega u \cdot v \, \mathrm{d}x.$$

Note that $a(\cdot, \cdot; \mu)$ depends on the parameter vector μ whereas $m(\cdot, \cdot)$ and Ω do not.

Remark 2.1. The equations of linear elasticity are used as a model problem as we are interested in the applications of vibro-acoustics. However this does not pose any restriction to the theoretical results shown in the following. Thus we could replace $a(\cdot, \cdot, \mu)$ by any H^1 elliptic bilinear form.

Let $V \subset \{v \in (H^1(\Omega))^d \mid v|_{\Gamma_D} = 0\}$ be a fixed conforming finite element space of dimension $\mathcal{N}$. Then, the discrete variational formulation of (2.1) reads as: Find the eigenvalues $\lambda(\mu) \in \mathbb{R}$ and the eigenfunctions $u(\mu) \in V$ such that

$$a(u(\mu), v; \mu) = \lambda(\mu)m(u(\mu), v) \quad \forall \, v \in V \tag{2.2}$$

for given $\mu \in \mathcal{P}$. We assume that the eigenvalues are positive and numbered as

$$0 < \lambda_1(\mu) \leq \ldots \leq \lambda_{\mathcal{N}}(\mu).$$

The corresponding eigenfunctions are denoted by $u_i(\mu) \in V$ for $i = 1, \ldots, \mathcal{N}$ with the normalization

$$m(u_i(\mu), u_j(\mu)) = \delta_{ij} \text{ for } 1 \leq i, j \leq \mathcal{N}.$$

In the present context, the error of the finite element solution is assumed to be very small. This is achieved by a fine mesh size leading to a large dimension $\mathcal{N}$. The discretization error analysis can be found, *e.g.*, in [1–3].

Let $L \geq 1$ be the number of distinct eigenvalues of (2.2). For multiple eigenvalues, we use the standard notation from [2] and denote the lowest index of the ith distinct eigenvalue by k_i and its multiplicity by q_i, $i = 1, \ldots, L$. We write $\mathcal{K}_i := \{k_i, \ldots, k_i + q_i - 1\}$. (Here and in the following, the dependency of the index notations on μ is suppressed as it is always clear from the context.) The corresponding eigenspaces are denoted by

$$U_i(\mu) := \mathrm{span}\left\{u_{k_i}(\mu), \ldots, u_{k_i+q_i-1}(\mu)\right\}.$$

Now, the goal is to find a computationally inexpensive but accurate surrogate model that can be used in the many-query or real-time context.

2.2. Model reduction

We consider a variational approximation of the μEVP in an N-dimensional reduced space

$$V_N := \mathrm{span}\left\{\zeta_n \mid n = 1, \ldots, N\right\} \subset V, \tag{2.3}$$

$N \ll \mathcal{N}$. As a matter of fact, the choice of V_N highly depends on the algorithmic methodology. Several (snapshot-based) possibilities are investigated in Section 4.

Now, the "reduced eigenvalue problem" reads as

$$(u_{\mathrm{red}}(\mu), \lambda_{\mathrm{red}}(\mu)) \in V_N \times \mathbb{R}, \quad a(u_{\mathrm{red}}(\mu), v; \mu) = \lambda_{\mathrm{red}}(\mu) m(u_{\mathrm{red}}(\mu), v) \quad \forall\, v \in V_N \tag{2.4}$$

for given $\mu \in \mathcal{P}$. Let us emphasize that all eigenpairs of interest are approximated in the same space V_N. As before, we assume a numbering $\lambda_{\mathrm{red},i}(\mu)$, $i = 1, \ldots, N$ of the "reduced eigenvalues". The minimum-maximum principles guarantee that $\lambda_i(\mu) \leq \lambda_{\mathrm{red},i}(\mu)$ for $i = 1, \ldots, N$; see ([2], Sect. 8). Note that the multiplicity of the finite element eigenvalues is not necessarily reflected in the reduced basis eigenvalues. The corresponding eigenfunctions are denoted by $u_{\mathrm{red},i}(\mu) \in V_N$ for $i = 1, \ldots, N$, again with the normalization

$$m(u_{\mathrm{red},i}(\mu), u_{\mathrm{red},j}(\mu)) = \delta_{ij} \text{ for } 1 \leq i, j \leq N.$$

In practice, as mentioned before, one is only interested in the first K eigenvalues for any chosen parameter. We expect that the dimension N required to achieve a certain accuracy will depend not only on the smoothness of the parameter-dependency of the μPDE but also on the number of outputs K.

In the present context, $a(\cdot, \cdot; \mu)$ is affine w. r. t. the parameter μ, *i.e.*,

$$a(u, v; \mu) = \sum_{q=1}^{Q} \Theta_q(\mu) a_q(u, v) \tag{2.5}$$

for suitable parameter-independent bilinear forms $a_q : (H^1(\Omega))^d \times (H^1(\Omega))^d \to \mathbb{R}$ and coefficients $\Theta_q : \mathcal{P} \to \mathbb{R}$, which are readily derived from the constitutive equations. For instance, we have two terms per subdomain in the isotropic case. This leads to a fast online evaluation as the cost of the assembly of the parameter-dependent reduced systems (*i.e.*, matrices in $\mathbb{R}^{N \times N}$ associated with (2.4)) is independent of $\mathcal{N}$. Note that the expansion (2.5) will also be exploited for an online-offline decomposition of the error estimators.
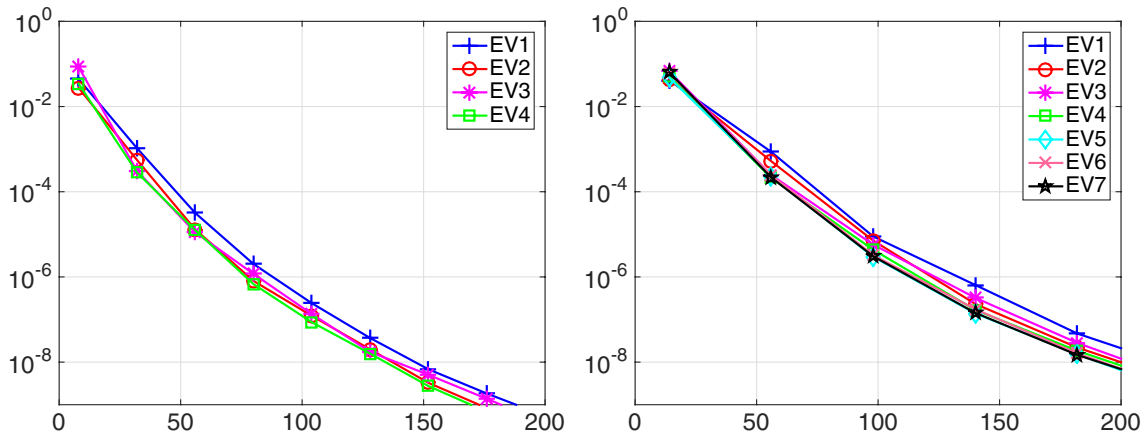
FIGURE 1. Convergence of POD methods ($0 < N \leq 200$) for the described μEVP with different numbers of outputs of interest: Average relative errors in the eigenvalues $\lambda_1, \ldots, \lambda_K$ for $K = 4$ (*left*) and $K = 7$ (*right*).

2.3. Model reduction by proper orthogonal decomposition

Before turning to the development of greedy methods and *a posteriori* error estimators for μPDEs, we illustrate the potential of model reduction techniques in the context of parameter dependent eigenvalue problems with multiple output values.

A common technique for model reduction is the proper orthogonal decomposition (POD) [21, 33, 34], which yields the best possible reduced space (in the sense that, for a given series of snapshots, the projection error w.r.t. the L^2-norm is minimized). To this end, let $S \subset V$ be a set of snapshots generated by solving the μ EVP (each time for K eigenfunctions) for all parameters in a sufficiently large training set $\Xi_{\text{train}}^{\text{POD}}$. Then, in the definition (2.3) of the reduced space V_N, orthonormalized functions $\{\zeta_1, \ldots, \zeta_N\} \subset \text{span}\{S\}$ are selected such that

$$\sum_{v \in S} \|v - \Pi_N v\|_{L^2(\Omega)}^2$$

is minimal. Here, Π_N is the L^2-orthogonal projection to $\text{span}\{\zeta_1, \ldots, \zeta_N\}$. This essentially amounts to assembling the $\#S \times \#S$-correlation matrix of the snapshots in S w.r.t. $(\cdot, \cdot)_{L^2(\Omega)}$ and finding its N largest eigenvalues and corresponding eigenvectors. For a detailed description of the usage of POD methods in the present context, see, *e.g.*, [21, 33, 34].

Figure 1 illustrates the convergence of the POD method for $K = 4$ and $K = 7$. (The details of the underlying numerical experiment are elaborated in Sect. 5). On the one hand, the results show that it is in principle possible to construct one single reduced space that effectively captures the parameter-dependent behavior of the first K eigenfunctions simultaneously. On the other hand, it is evident that the RB dimension N required for a certain accuracy increases with K. More precisely, the asymptotic decay of the error is approximately $C_4 \, e^{-0.0513 \cdot N}$ for $K = 4$ and $C_7 \, e^{-0.0477 \cdot N}$ for $K = 7$ for some constants C_4 and C_7. From the cost point of view the POD is quite expensive, and thus we focus on computationally efficient greedy strategies in combination with *a posteriori* error bounds.

3. A POSTERIORI ERROR ESTIMATION

In this section, we will establish *a posteriori* error estimators for our output quantities, in this case the eigenvalues. It is important to determine such estimators in order to find out which basis functions should be

selected by the greedy method. In particular, their computational evaluation must only depend on the basis size N but not on the dimension of the finite element space $\mathcal{N}$.

To this end, we first derive error bounds that still depend on the finite element eigenvalues, and in particular on their multiplicities. Then, Section 3.2 is devoted to a computable approximation yielding the desired error estimators.

In Section 3.3, we recall a standard online-offline decomposition.

3.1. Error bounds

Let the parameter-dependent energy norm be defined as $\|\cdot\|_{\mu;V} := a(\cdot,\cdot;\mu)^{\frac{1}{2}}$. In addition to a parameter dependent norm we are using a parameter indepenent norm defined as $\|\cdot\|_{\hat{\mu};V} := \hat{a}(\cdot,\cdot)^{\frac{1}{2}} := a(\cdot,\cdot;\hat{\mu})^{\frac{1}{2}}$. For a linear functional $r : V \to \mathbb{R}$, we define the corresponding dual norms by

$$\|r\|_{\mu;V'} := \sup_{0 \neq v \in V} \frac{r(v)}{\|v\|_{\mu;V}} \quad \text{and} \quad \|r\|_{\hat{\mu};V'} := \sup_{0 \neq v \in V} \frac{r(v)}{\|v\|_{\hat{\mu};V}},$$

respectively.

The analysis and the practical implementation employ different error representations, namely the so-called reconstructed errors w. r. t. the bilinear forms $a(\cdot,\cdot;\mu)$ and $\hat{a}(\cdot,\cdot)$. Using the residual

$$v \mapsto r_i(v;\mu) := a(u_{\mathrm{red},i}(\mu), v; \mu) - \lambda_{\mathrm{red},i}(\mu)\, m(u_{\mathrm{red},i}(\mu), v)$$

for $i = 1, \ldots, N$, we define $e_i(\mu) \in V$ and $\hat{e}_i(\mu) \in V$ by

$$a(e_i(\mu), v; \mu) = r_i(v; \mu) \quad \forall\, v \in V \tag{3.1}$$

and

$$\hat{a}(\hat{e}_i(\mu), v) = r_i(v; \mu) \quad \forall\, v \in V, \tag{3.2}$$

respectively. In particular, $\|r_i(\cdot;\mu)\|_{\mu;V'} = \|e_i(\mu)\|_{\mu;V}$ and $\|r_i(\cdot;\mu)\|_{\hat{\mu};V'} = \|\hat{e}_i(\mu)\|_{\hat{\mu};V}$.

For any $\mu \in \mathcal{P}$, assume that $g(\mu) > 0$ is a generalized coercivity constant such that $g(\mu)\hat{a}(v,v) \leq a(v,v;\mu)$ for all $v \in V$. (Technically speaking, $g(\mu)$ is the parameter-dependent coercivity constant of $a(\cdot,\cdot;\mu)$ w. r. t. $\|\cdot\|_{\hat{\mu};V}$.) This implies

$$\|r\|_{\mu;V'} \leq g(\mu)^{-\frac{1}{2}} \|r\|_{\hat{\mu};V'} \tag{3.3}$$

for any $r \in V'$ and

$$\|v\|_{\hat{\mu};V} \leq g(\mu)^{-\frac{1}{2}} \|v\|_{\mu;V} \tag{3.4}$$

for any $v \in V$.

We are now ready to prove the error bounds. The following theorem, combined with the computational/algorithmic aspects in Section 3.2 and Section 4, generalize the results of [24, 30–32] for the case of multiple eigenvalues.

Theorem 3.1. *Let* $1 \leq i \leq L$ *such that* $k_i + q_i - 1 \leq N$. *For* $j = 1, \ldots, q_i$, *set*

$$\tilde{d}_{k_i+j-1}(\mu) := \min_{\mathcal{N} \geq l > k_i+q_i-1} \left| \frac{\lambda_l(\mu) - \lambda_{\mathrm{red},\,k_i+j-1}(\mu)}{\lambda_l(\mu)} \right|. \tag{3.5}$$

Then,

$$0 \le \lambda_{\text{red}, k_i+j-1}(\mu) - \lambda_{k_i}(\mu) \le \frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\tilde{d}_{k_i+j-1}(\mu)} \left(1 + \frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}}{\tilde{d}_{k_i+j-1}(\mu)^2 \sqrt{\lambda_{k_i}}}\right). \tag{3.6}$$

Proof. Fix $\mu \in \mathcal{P}$, $1 \le i \le L$ and $1 \le j \le q_i$. Let $u_{\text{red}, k_i+j-1}(\mu) = \sum_{l=1}^{\mathcal{N}} \alpha_l u_l(\mu)$ and $e_{k_i+j-1}(\mu) = \sum_{l=1}^{\mathcal{N}} \beta_l u_l(\mu)$. By (3.1), we find

$$\beta_l = \alpha_l \frac{\lambda_l(\mu) - \lambda_{\text{red}, k_i+j-1}(\mu)}{\lambda_l(\mu)}.$$

Therefore, we get

$$\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2 = \sum_{l=1}^{\mathcal{N}} \alpha_l^2 \left(\frac{\lambda_l(\mu) - \lambda_{\text{red}, k_i+j-1}(\mu)}{\lambda_l(\mu)}\right)^2 \lambda_l(\mu)$$

$$\ge \sum_{l>k_i+q_i-1} \alpha_l^2 \left(\frac{\lambda_l(\mu) - \lambda_{\text{red}, k_i+j-1}(\mu)}{\lambda_l(\mu)}\right)^2 \lambda_l(\mu)$$

$$\ge \tilde{d}_{k_i+j-1}(\mu)^2 \sum_{l>k_i+q_i-1} \alpha_l^2 \lambda_l(\mu). \tag{3.7}$$

Using the fact that $\sum_l \alpha_l^2 = 1$, $\lambda_l(\mu) \le \lambda_{k_i}(\mu)$ for $l \le k_i+q_i-1$, we find for the difference between approximated and detailed eigenvalue

$$\Delta\lambda_{k_i} := \lambda_{\text{red}, k_i+j-1}(\mu) - \lambda_{k_i}(\mu) = a(u_{\text{red}, k_i+j-1}(\mu), u_{\text{red}, k_i+j-1}(\mu); \mu) - \lambda_{k_i}(\mu)$$

$$= \sum_{l=1}^{\mathcal{N}} \alpha_l^2 \lambda_l(\mu) - \lambda_{k_i}(\mu)$$

$$= \sum_{l \le k_i+q_i-1} \alpha_l^2 (\lambda_l(\mu) - \lambda_{k_i}(\mu)) + \sum_{l>k_i+q_i-1} \alpha_l^2 (\lambda_l(\mu) - \lambda_{k_i}(\mu))$$

$$\le \sum_{l>k_i+q_i-1} \alpha_l^2 (\lambda_l(\mu) - \lambda_{k_i}(\mu)). \tag{3.8}$$

From this we obtain two upper bounds for $\Delta\lambda_{k_i}$. The first one follows trivially from the fact that $\lambda_{k_i}(\mu) > 0$ and (3.7)

$$\Delta\lambda_{k_i} \le \sum_{l>k_i+q_i-1} \alpha_l^2 \lambda_l(\mu) \le \frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\tilde{d}_{k_i+j-1}(\mu)^2}. \tag{3.9}$$

The second bound is based on the Cauchy–Schwarz inequality and on Young's inequality. In terms of,

$$(\lambda_l(\mu) - \lambda_{k_i}(\mu))^2 \le (1+\epsilon)(\lambda_l(\mu) - \lambda_{\text{red}, k_i+j-1}(\mu))^2 + \left(1 + \frac{1}{\epsilon}\right)(\Delta\lambda_{k_i})^2$$

for $\epsilon > 0$, we get from (3.7) and (3.8)

$$
\begin{aligned}
\Delta\lambda_{k_i} &= \sum_{l>k_i+q_i-1} \alpha_l \frac{\lambda_l(\mu) - \lambda_{k_i}(\mu)}{\lambda_l(\mu)} \sqrt{\lambda_l(\mu)} \alpha_l \sqrt{\lambda_l(\mu)} \\
&\leq \left(\sum_{l>k_i+q_i-1} \alpha_l^2 (\frac{\lambda_l(\mu) - \lambda_{k_i}(\mu)}{\lambda_l(\mu)})^2 \lambda_l(\mu) \right)^{\frac{1}{2}} \left(\sum_{l>k_i+q_i-1} \alpha_l^2 \lambda_l(\mu) \right)^{\frac{1}{2}} \\
&\leq \frac{1}{\tilde{d}_{k_i+j-1}(\mu)} \|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'} \sqrt{(1+\epsilon) \|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2 + (1+\frac{1}{\epsilon})\Delta\lambda_{k_i}^2 \sum_{l>k_i+q_i-1} \alpha_l^2 \frac{1}{\lambda_l(\mu)}} \\
&\leq \frac{1}{\tilde{d}_{k_i+j-1}(\mu)} \|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'} \sqrt{(1+\epsilon) \|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2 + (1+\frac{1}{\epsilon})\frac{\Delta\lambda_{k_i}^2}{\lambda_{k_i}(\mu)}} \\
&\leq \frac{1}{\tilde{d}_{k_i+j-1}(\mu)} \|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2 \sqrt{1+\epsilon + (1+\frac{1}{\epsilon})\frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\tilde{d}_{k_i+j-1}(\mu)^4 \lambda_{k_i}(\mu)}}.
\end{aligned}
$$

Setting $\epsilon = \frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}}{\tilde{d}_{k_i+j-1}(\mu)^2 \sqrt{\lambda_{k_i}(\mu)}}$ gives the upper bound in (3.6). The lower bound follows directly from ([2], Sect. 8). □

Besides the generalization to multiple eigenvalues, let us point out that our bounds are sharper than the ones (*e.g.*, [32], Prop. 1), as the lowest order term in (3.6) is of the form $\frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\tilde{d}_{k_i+j-1}(\mu)}$ rather than $\frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\tilde{d}_{k_i+j-1}(\mu)^2}$. Note that the error bounds in Theorem 3.1 still depend on the finite element solution *via* the eigenvalues $\lambda_l(\mu)$ in (3.5).

Remark 3.2. It is also possible to give an upper bound for the eigenvectors by replacing $\tilde{d}_{k_i+j-1}(\mu)$ by $\hat{d}_{k_i+j-1}(\mu)$ defined as

$$
\hat{d}_{k_i+j-1}(\mu) := \min_{\mathcal{N} \geq l > k_i+q_i-1 \vee l < k_i} \left| \frac{\lambda_l(\mu) - \lambda_{\mathrm{red},\, k_i+j-1}(\mu)}{\lambda_l(\mu)} \right|.
$$

Using now $\Pi_i : V \to U_i(\mu)$ as the orthogonal projection w.r.t. the L^2-inner product, we define $\bar{v} := \Pi_i(u_{\mathrm{red},\, k_i+j-1}(\mu)) = \sum_{\mathcal{N} \geq l > k_i+q_i-1 \vee l < k_i} \alpha_l u_l(\mu)$ and give the upper bound as

$$
\|u_{\mathrm{red},\, k_i+j-1}(\mu) - \bar{v}\|_{\mu;V}^2 = \left\| \sum_{\mathcal{N} \geq l > k_i+q_i-1 \vee l < k_i} \alpha_l u_l(\mu) \right\|_{\mu;V}^2 = \sum_{\mathcal{N} \geq l > k_i+q_i-1 \vee l < k_i} \alpha_l^2 \lambda_l(\mu) \leq \frac{\|r_{k_i+j-1}(\cdot;\mu)\|_{\mu;V'}^2}{\hat{d}_{k_i+j-1}(\mu)^2}.
$$

3.2. Error estimators

We now derive approximate error bounds that are computable in the sense that they do not depend on the finite element solution. To achieve this, it remains to approximate $\tilde{d}_i$, which may be interpreted as a measure for the relative distance between neighboring/adjacent eigenvalues, particularly to decide which of the indices to exclude from the minimum. We point out that the dimension of the (detailed) eigenspace is not accessible. The application scenario we have in mind features multiple eigenvalues with their multiplicities depending on the parameter. It is therefore impossible to determine the structure of the spectrum (*i.e.*, the indices k_i or the index sets $\mathcal{K}_i$) *a priori*.

Recall that the first K eigenvalues are the output quantities of interest. Assume that the reduced basis method converges in the following sense: for $\mu \in \mathcal{P}$ and $1 \leq i \leq K$,

$$\lambda_{\text{red},\,i}(\mu) \to \lambda_i(\mu) \text{ for } N \to \mathcal{N}.$$

In particular, $\lambda_{\text{red},\,j}(\mu) \to \lambda_{k_i}(\mu)$ for $N \to \mathcal{N}$ for $j \in \mathcal{K}_i$.

Given the eigenvalues $\lambda_{\text{red},\,i}(\mu)$, $i = 1, \ldots, K$, of (2.4), we replace $\lambda_l(\mu)$ in (3.5) by $\lambda_{\text{red},\,l}(\mu)$ and approximate $\mathcal{K}_i$ by

$$\mathcal{K}_{\text{red},\,i} := \left\{ 1 \leq j \leq K + \mathfrak{r}; \ \left| \frac{\lambda_{\text{red},\,j}(\mu) - \lambda_{\text{red},\,i}(\mu)}{\lambda_{\text{red},\,j}(\mu)} \right| < \varepsilon_\lambda \right\}$$

for a chosen "tolerance" $\varepsilon_\lambda > 0$ and with $\mathfrak{r}$ as the difference between the index of the first eigenvalue after the multiplicity of the Kth eigenvalue and the Kth eigenvalue itself. In the case that we know *a priori* the maximal multiplicity of all relevant eigenvalues for all parameters, we set $\mathfrak{r}$ equal to this value. Otherwise we select it adaptively during the initialization phase of the greedy method. More precisely, we start with $\mathfrak{r} = 1$ and increase it by one as long as $K + \mathfrak{r} \in \mathcal{K}_{\text{red},\,i}$. Thus $\#\mathcal{K}_{\text{red},\,i}$ will be our best guess for the multiplicity of the eigenvalue to which $\lambda_{\text{red},\,i}(\mu)$ converges. Then for $1 \leq i \leq K$,

$$d_i(\mu) := \min_{\substack{l \notin \mathcal{K}_{\text{red},\,i} \\ K+\mathfrak{r} \geq l > i}} \left| \frac{\lambda_{\text{red},\,l}(\mu) - \lambda_{\text{red},\,i}(\mu)}{\lambda_{\text{red},\,l}(\mu)} \right| \tag{3.10}$$

is the relative distance of $\lambda_{\text{red},\,i}(\mu)$ to the reduced eigenvalues that are further away than the chosen tolerance ε_λ. The adaptive selection of $\mathfrak{r}$ guarantees that even for $i = K$ and multiple eigenvalues $d_i(\mu)$ is easily computable and does not severely underestimate $\tilde{d}_i(\mu)$.

Finally, since we are looking for an asymptotic estimator for the relative error in the eigenvalues which is cheaply computable in the online-phase, we neglect the higher order term in (3.6). In addition, the parameter-dependent norm $\|\cdot\|_{\mu;V'}$ is replaced by the parameter-independent norm $\|\cdot\|_{\hat{\mu};V'}$ by (3.3), which introduces an additional factor $g(\mu)^{-1}$. To summarize we can state the following corollary:

Corollary 3.3. *Let $i = 1, \ldots, K$ and $\lambda_{\text{red},\,i}(\mu) \to \lambda_i(\mu)$ for $N \to \mathcal{N}$. Furthermore let $\mathcal{K}_{\text{red},\,i}$ be defined as above and the distance between neighboring eigenvalues $d_i(\mu)$ be given as in (3.10). Then the error estimator given by*

$$\mu \mapsto \eta_i(\mu) := \frac{\|r_i(\cdot;\mu)\|^2_{\hat{\mu};V'}}{g(\mu) \cdot d_i(\mu) \cdot \lambda_{\text{red},\,i}(\mu)}. \tag{3.11}$$

is asymptotically reliable in the sense that

$$0 \leq \frac{\lambda_{\text{red},\,k_i+j-1}(\mu) - \lambda_{k_i}(\mu)}{\lambda_{k_i}} \leq C\eta_i(\mu),$$

with C tending to one as N tends to $\mathcal{N}$.

Note that the approximation in (3.10) is, in general, less accurate for $i = K$. This is because the space V_N is built to approximate well the K outputs, but for the Kth estimator we need the $(K + \mathfrak{r})$th outputs with $\mathfrak{r} \geq 1$, which are approximated only roughly. The tolerance ε_λ has to be selected such that it reflects the desired accuracy of the RB approximation.

3.3. Online-offline decomposition

All error estimator contributions may be decomposed as already outlined in [24]. Let $(\zeta_n)_{1 \leq n \leq N}$ be the orthonormal basis (w.r.t. $m(\cdot, \cdot)$) of V_N. For $0 \leq q, p \leq Q$ let $\hat{A}^{q,p} \in \mathbb{R}^{N \times N}$ with $\hat{A}^{q,p}_{n,m} := \hat{a}(\xi^q_n, \xi^p_m)$ for $1 \leq n, m \leq N$ where

$$\hat{a}(\xi^q_n, v) = a_q(\zeta_n, v), \quad \forall v \in V, \ 1 \leq n \leq N, \ 1 \leq q \leq Q, \tag{3.12}$$

and

$$\hat{a}(\xi_n^0, v) = m(\zeta_n, v), \quad \forall\, v \in V,\ 1 \le n \le N. \tag{3.13}$$

In the following, we identify the function $u_{\mathrm{red},\,i}(\mu) \in V_N$ and its vector representation w. r. t. the basis $(\zeta_n)_{1 \le n \le N}$ such that $(u_{\mathrm{red},\,i}(\mu))_n$ denotes the nth coefficient. Then, given a reduced eigenpair $(u_{\mathrm{red},\,i}(\mu), \lambda_{\mathrm{red},\,i}(\mu))$, we have the error representation

$$\hat{e}_i(\mu) = \sum_{n=1}^{N} \sum_{q=1}^{Q} \Theta_q(\mu)\,(u_{\mathrm{red},\,i}(\mu))_n\, \xi_n^q - \lambda_{\mathrm{red},\,i}(\mu) \sum_{n=1}^{N} (u_{\mathrm{red},\,i}(\mu))_n\, \xi_n^0$$

by (3.2). Consequently, the main contribution of $\eta_i(\mu)$ decomposes into

$$
\begin{aligned}
\|r_i(\cdot;\mu)\|_{\hat{\mu};V'}^2 = & \sum_{n=1}^{N} \sum_{m=1}^{N} \sum_{q=1}^{Q} \sum_{p=1}^{Q} (u_{\mathrm{red},\,i}(\mu))_n\, (u_{\mathrm{red},\,i}(\mu))_m\, \Theta_q(\mu)\Theta_p(\mu)\, \hat{A}_{n,m}^{q,p} \\
& + \lambda_{\mathrm{red},\,i}^2(\mu) \sum_{n=1}^{N} \sum_{m=1}^{N} (u_{\mathrm{red},\,i}(\mu))_n\, (u_{\mathrm{red},\,i}(\mu))_m\, \hat{A}_{n,m}^{0,0} \\
& - 2\,\lambda_{\mathrm{red},\,i}(\mu) \sum_{n=1}^{N} \sum_{m=1}^{N} \sum_{q=1}^{Q} (u_{\mathrm{red},\,i}(\mu))_n\, (u_{\mathrm{red},\,i}(\mu))_m\, \Theta_q(\mu)\, \hat{A}_{n,m}^{q,0}.
\end{aligned}
$$

We recall that only a single reduced space is built for the approximation of all eigenvectors simultaneously. Thus the above decomposition uses the same offline ingredients for all $1 \le i \le K$. In particular, the number K of desired eigenpairs does not directly influence the complexity (only *via* the reduced space dimension N).

4. Algorithms / Basis construction

In this section, we present different greedy strategies that employ the error estimators of Section 3 to build the reduced space in equation (2.3). The advantage as compared to the POD method motivated in Section 2.3 is that only relatively few finite element solutions of the μEVP need to be computed.

Since we use a single space for the approximation of multiple outputs, we have several natural possibilities which are investigated in Section 4.1. Section 4.2 is devoted to an extension that takes into account multiple eigenvalues. In Section 4.3, a remedy for the potential unreliability of the error estimators for small N is discussed.

4.1. Greedy selection of snapshots for single eigenvalues

Recall that the K smallest eigenvalues are the quantities of interest, where K is typically $2-20$ for our application scenario. In principle, given a reduced space, one could try to identify a suitable $\mu \in \mathcal{P}$ and then include the first K eigenfunctions for this parameter value. (In each greedy step, this would require the detailed FE solution of (2.2) for one parameter only.) However, numerical studies clearly show that this naive choice is far from optimal as the generated reduced spaces tend to be much too large. This is because the errors in the individual eigenvalues and eigenfunctions are only very weakly correlated, if at all. There are at least the following two much more natural options.

Let a sufficiently rich training set $\Xi_{\mathrm{train}} \subset \mathcal{P}$ be given. Then, in Algorithm 1, the individual $\arg\max$ for each $1 \le i \le K$ is chosen separately. In contrast, Algorithm 2 chooses only one single $\arg\max$. Note that both Algorithm 1 (line 7) and Algorithm 2 (line 6) require the evaluation of all error estimators at all parameters in Ξ_{train} to determine the choice of μ. This does not lead to large computations since the calculations are only performed with the reduced space of size N, such that we obtain K reduced eigenpairs for any $\mu \in \Xi_{\mathrm{train}}$, see also Section 3.3. However, Algorithm 1 (line 10) and Algorithm 2 (line 8) require also finite element solutions which then determine the reduced basis space.

Algorithm 1. Multi-choice greedy.

1: **for** $i = 1, \ldots, K$ **do**
2:     $\zeta_i \leftarrow u_i(\hat{\mu})$
3: **end for**
4: $N \leftarrow K$
5: **while** $N < N_{\max}$ **do**
6:     **for** $i = 1, \ldots, K$ **do**
7:         $\mu_{\max,i} \leftarrow \arg\max_{\mu \in \Xi_{\mathrm{train}}} \eta_i(\mu)$
8:         **if** $\eta_i(\mu_{\max,i}) > \varepsilon_{\mathrm{tol}}$ **then**
9:             $N \leftarrow N + 1$
10:             $\zeta_N \leftarrow u_i(\mu_{\max,i})$ (orthonormalized)
11:         **end if**
12:     **end for**
13:     **if** $\max_{\mu \in \Xi_{\mathrm{train}}, 1 \leq i \leq K} \eta_i(\mu) < \varepsilon_{\mathrm{tol}}$ **then**
14:         **break**
15:     **end if**
16: **end while**

Algorithm 2. Single-choice greedy.

1: **for** $i = 1, \ldots, K$ **do**
2:     $\zeta_i \leftarrow u_i(\hat{\mu})$
3: **end for**
4: $N \leftarrow K$
5: **while** $N < N_{\max}$ **do**
6:     $(\mu_{\max}, i_{\max}) \leftarrow \arg\max_{\mu \in \Xi_{\mathrm{train}}, 1 \leq i \leq K} \eta_i(\mu)$
7:     $N \leftarrow N + 1$
8:     $\zeta_N \leftarrow u_{i_{\max}}(\mu_{\max})$ (orthonormalized)
9:     **if** $\max_{\mu \in \Xi_{\mathrm{train}}, 1 \leq i \leq K} \eta_i(\mu) < \varepsilon_{\mathrm{tol}}$ **then**
10:         **break**
11:     **end if**
12: **end while**

The multi-choice variant rests on the intuition that the individual eigenfunctions can/should be approximated separately. In contrast, the single-choice variant takes into account that the approximation power of eigenfunctions to large eigenvalues can be exploited also for eigenfunctions to smaller eigenvalues.

During the greedy procedure, we orthonormalize the selected basis functions. Not only does this yield small condition numbers of the reduced systems; it is also beneficial for the special treatment of multiple eigenvalues described in the next section.

In Algorithm 1 (line 10) and Algorithm 2 (line 8), an orthonormalization is performed. For this purpose, let $\Pi_N : V \rightarrow V_N$ be the L^2-orthogonal projection to the current reduced space. For a snapshot candidate $\zeta \in V$ (*i.e.*, one of the eigenfunctions chosen as described above), we compute $\tilde{\zeta} := \zeta - \Pi_N \zeta$. Then, if $\|\tilde{\zeta}\|$ is sufficiently large ($\geq \varepsilon_{\mathrm{proj}}$), the "new contribution" $\frac{\tilde{\zeta}}{\|\tilde{\zeta}\|}$ is included in the reduced basis; see also Section 4.2.

4.2. Extended selection for multiple eigenvalues

In case of multiple eigenvalues, the greedy method needs to be modified as follows. Assume an index $1 \leq \tilde{\imath} \leq K$ and a parameter $\tilde{\mu}$ have been selected by means of the eigenvalue-based estimators $(\eta_i)_{i=1,\ldots,K}$, in Algorithm 2 (line 6), or several parameters $\tilde{\mu}$ have been selected in Algorithm 1 (line 7), such that the span of $u_{\tilde{\imath}}(\tilde{\mu})$ is to be included in the reduced space.

However, a large value of $\eta_{\tilde{i}}(\tilde{\mu})$ merely indicates that the corresponding (fine) eigenspace $U_{\tilde{i}}(\tilde{\mu})$ contains functions that are badly approximated by the current reduced space. Nevertheless the eigenspace might also contain other functions that are already well approximated. Consequently, if the detailed eigenvalue associated with a chosen snapshot has multiplicity greater than one, we aim to add all the eigenfunctions for the multiple eigenvalue, except the ones which are already approximated well enough. A motivation for exploring the whole eigenspace for multiple eigenvalues is to guarantee that we take the correct eigenvalue/eigenfunction, since we cannot ensure that the indexed eigenvalue/eigenfunction in the reduced space is the same as in the detailed calculation. This is due to the fact that there is no prescribed ordering for the eigenfunctions corresponding to a multiple eigenvalue.

As for the definition of $d_i(\mu)$ one has to compute a sufficient number $K' > K$ of eigenfunctions of the finite element problem μEVP (2.2) such that $\lambda_{K'}(\tilde{\mu})/\lambda_K(\tilde{\mu}) > 1 + \varepsilon_\lambda$. Then, lines 9–10 in Algorithm 1 are replaced by:

---

for all $j \geq 1$ **with** $|\lambda_j(\mu_{\max,i}) - \lambda_i(\mu_{\max,i})|/\lambda_i(\mu_{\max,i}) \leq \varepsilon_\lambda$ **do**
    if $\|u_j(\mu_{\max,i}) - \Pi_N u_j(\mu_{\max,i})\|_{L^2(\Omega)} \geq \varepsilon_{\text{proj}}$ **then**
        $N \leftarrow N + 1$
        $\zeta_N \leftarrow u_j(\mu_{\max,i})$ (orthonormalized)
    end if
end for

Analogously, lines 7–8 in Algorithm 2 now read as:

---

for all $j \geq 1$ **with** $|\lambda_j(\mu_{\max}) - \lambda_{i_{\max}}(\mu_{\max})|/\lambda_{i_{\max}}(\mu_{\max}) \leq \varepsilon_\lambda$ **do**
    if $\|u_j(\mu_{\max}) - \Pi_N u_j(\mu_{\max})\|_{L^2(\Omega)} \geq \varepsilon_{\text{proj}}$ **then**
        $N \leftarrow N + 1$
        $\zeta_N \leftarrow u_j(\mu_{\max})$ (orthonormalized)
    end if
end for

Here, $\Pi_N : V \to V_N$ denotes the L^2-orthogonal projection. The parameter $\varepsilon_{\text{proj}}$ is a small tolerance that prevents the selection of functions that are already approximated sufficiently well.

4.3. Initialization of the greedy method

In our calculations, we need an error estimator for the Kth eigenvalue. For the computation of this estimator, we need a rough approximation of the $(K + \mathfrak{r})$th eigenvalue. In order to ensure that our reduced space has the ability to roughly approximate this $(K + \mathfrak{r})$th eigenvalue, we use an initial approximation space in which we include the corresponding components. We suggest to include components using the proper orthogonal decomposition method described in Section 2.3 (with $N = N_{\text{init}}$) applied to a small number of snapshots. Here, the snapshots $S \subset V$ are associated with a training set $\Xi_{\text{train}}^{\text{POD}}$ typically of size 2^P, taking into account the extension described in Section 4.2. This initial approximation space of dimension N_{init}, which is constructed as an initialization step for the greedy algorithm, should be sufficiently large as the reliability of the error estimators analyzed in Section 3 can depend on the dimension of the reduced space. To make sure that we are able to calculate and to approximate the $(K + \mathfrak{r})$th eigenvalues, we chose our N_{init} to be at least $(K + \mathfrak{r})$ times a factor ≥ 1.5.

5. Numerical results

In this section, the performance of the proposed algorithms is illustrated by numerical examples, in two and in three dimensions. For the two-dimensional calculations we use plane strain elasticity while for the three-dimensional simulations we use linear elasticity. The implementation is performed in MATLAB based on the RBmatlab [10] library. We investigate the individual components and highlight their benefits in several steps.

5.1. Preliminaries

First, in Section 5.2 to Section 5.5, we choose Ω as rectangle of size 3.0×1.0 with Dirichlet boundary on the left and on the right. Let Ω be split into three subdomains of size 1.0×1.0. The material parameters E and ν used for these subdomains are in the range of $10-100$ and $0.1-0.4$, respectively; we have $P = 6$ and $Q = 6$. We choose a uniform random sample of size 10 000 as set of training parameters $\Xi_{\text{train}} \subset \mathcal{P}$. To evaluate the errors, another sufficiently rich set of parameters $\Xi_{\text{test}} \subset \mathcal{P}$ is used of size 1000. For our initial space we choose $N_{\text{init}} \leq 40$, depending on the desired number of eigenvalues K. We always report the average errors of the reduced approximations given by

$$\frac{1}{\#\Xi_{\text{test}}} \sum_{\mu \in \Xi_{\text{test}}} \frac{\lambda_{\text{red}, i}(\mu) - \lambda_i(\mu)}{\lambda_i(\mu)}$$

and comment on the standard deviation at the end of Section 5.4 in Remark 5.1.

For the generalized coercivity estimate, we exploit the affine decomposition of the bilinear form and set

$$g(\mu) := \min_{q=1,\ldots,Q} \frac{\Theta_q(\mu)}{\Theta_q(\hat{\mu})}. \tag{5.1}$$

We emphasize that $g(\mu)$ merely relates the bilinear forms $a(\cdot, \cdot; \mu)$ and $\hat{a}(\cdot, \cdot)$; a coercivity estimate for $a(\cdot, \cdot; \mu)$ itself is not required in the present context. Note that (5.1) indeed yields an admissible parameter-dependent constant provided the bilinear forms $a_q(\cdot, \cdot)$ in (2.5) are positive semi-definite and the coefficient functions $\Theta_q(\cdot)$ in (2.5) are positive; (see, *e.g.*, [29], Sect. 4.2.2). This is true for our application. Better results (*i.e.*, a larger lower bound) could be obtained by the more expensive successive constraint method [19]. In the present setting, the estimate (5.1) is typically smaller than the exact solution of the corresponding generalized eigenvalue problem by a factor ranging from 0.7 to 0.98.

5.2. Extended selection *vs.* non-extended selection

We first illustrate the necessity of the extended selection for multiple eigenvalues. Figure 2 shows the behavior of a POD method with (*left*) and without (*right*) the extended selection described in Section 4.2 for the first two eigenvalues ($K = 2$). Figure 3 shows the same comparison for the greedy method (Algorithm 2). For both the POD method and the greedy method, we observe that in the variants without extension the convergence for the second eigenvalue becomes slower after a certain number of basis functions has been included. In contrast, the extended selection yields convergence curves that approximately coincide.

The shortcomings of the non-extended methods may be explained by the fact that the second eigenvalue has multiplicity two for certain parameters and in these cases, for the multiple eigenvalue, the correct eigenfunction is not necessarily chosen. Note that the effect is more significant for a smaller POD training size (Fig. 2, second row) as it is less likely that all directions of an eigenspace are present in the snapshot set. The convergence of the second reduced eigenvalue possibly improves drastically if, incidentally, the missing component is added during the greedy method.

5.3. Multi-choice *vs.* single-choice greedy method

Here, we illustrate the benefit of Algorithm 2 in comparison to Algorithm 1. In Figure 4, for $K = 4$, one can see that with Algorithm 1 (*left*) the convergence behavior varies over the course of the greedy method while with Algorithm 2 (*right*) all desired eigenvalues exhibit similar convergence. (This also holds true for the errors in the eigenfunctions not shown here.)

The poor convergence of the third eigenvalue only improves rapidly at $N \approx 170$, after the other three eigenvalues have reached an accuracy in the order of the target tolerance, and thus the algorithm only chooses EV 3. This effect (namely an imbalanced resolution of the relevant eigenspaces during the greedy method) is directly related to the inappropriate *a priori* assumption of Algorithm 1 that roughly the same number
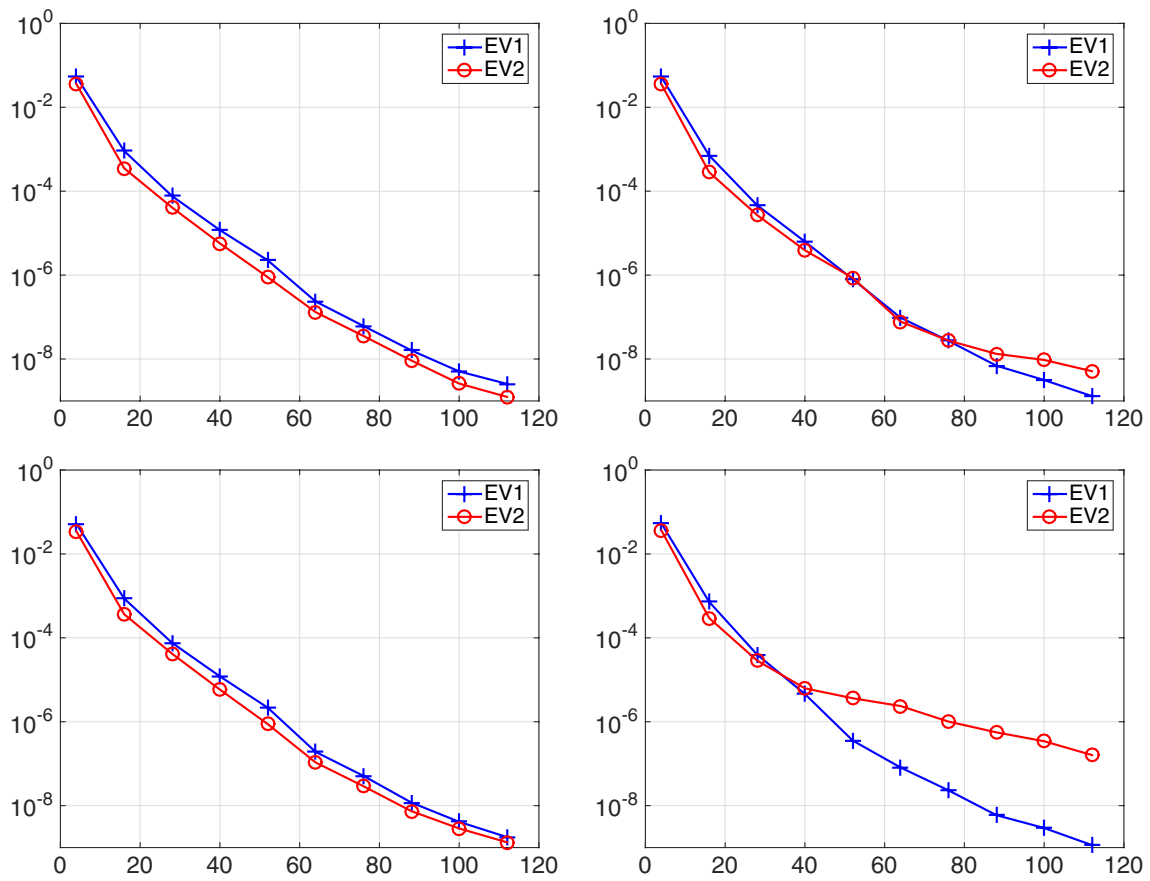
FIGURE 2. Error decay for the eigenvalues with the POD method: extended (*left*) *vs.* non-extended (*right*). *First row*: training size 10 000; *second row*: training size 1000.
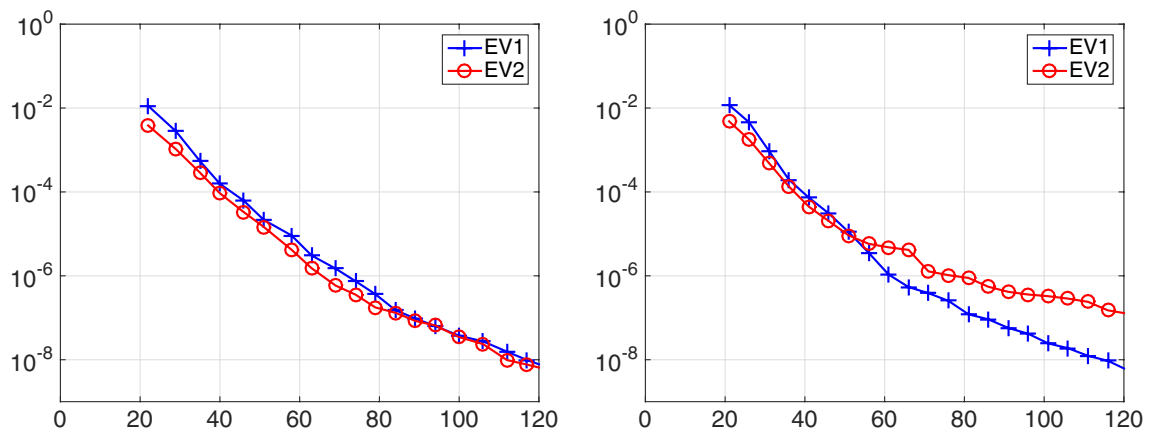

FIGURE 3. Error decay for eigenvalues with the greedy method: extended (*left*) *vs.* non-extended (*right*); training size 1000.
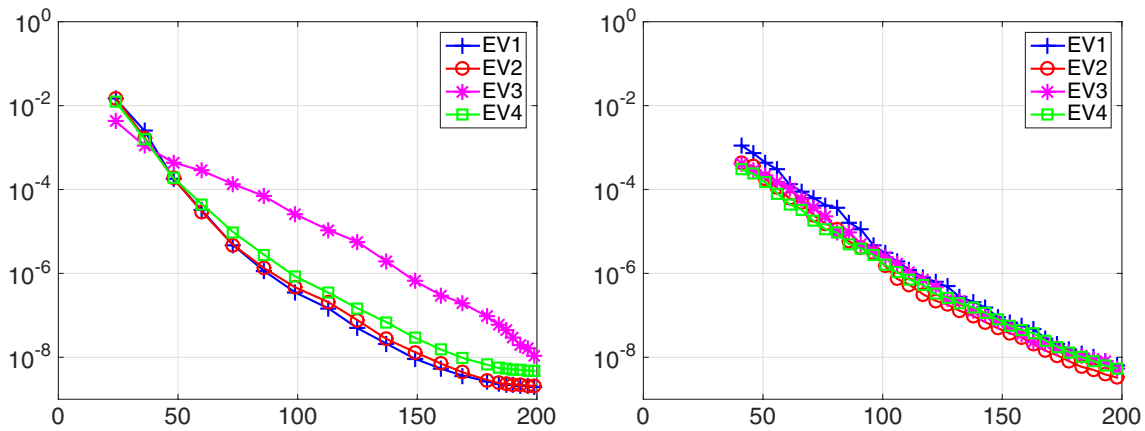
FIGURE 4. RB error decay: comparison of Algorithm 1 (*left*) and Algorithm 2 (*right*) for $K = 4$.
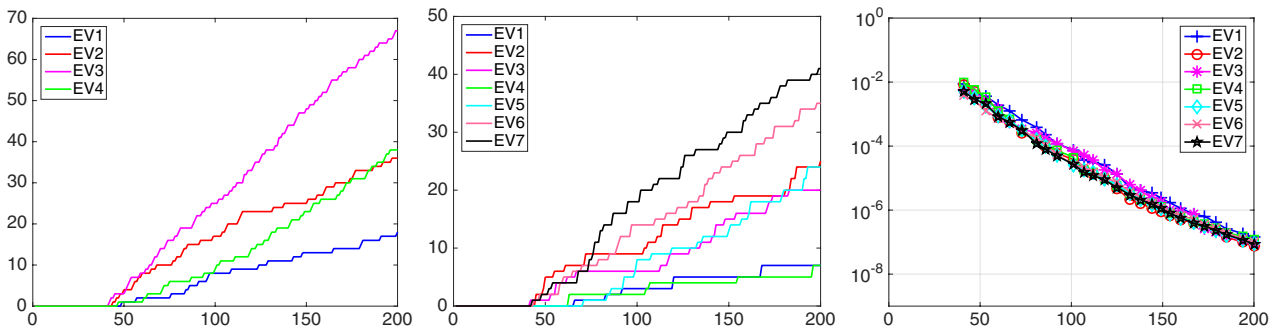

FIGURE 5. *Left and center*: accumulated numbers of chosen eigenfunctions over the course of Algorithm 2 for $K = 4$ and $K = 7$. *Right*: error decay for $K = 7$.

of snapshots corresponding to the first K eigenvalues should be included in the reduced space. At this point it should also be noted that in general Algorithm 1 creates a larger RB space than Algorithm 2 as soon as more eigenvalues have a poor convergence.

To further illustrate the behavior of the single-choice greedy method, in Figure 5, we report the accumulated numbers of chosen eigenfunctions corresponding to $\lambda_1, \ldots, \lambda_K$ over the course of Algorithm 2 for $K = 4$ (*left*) and $K = 7$ (*center*), as selected by the error estimators in line 6. The reason for the greedy algorithm not selecting any eigenfunctions before a basis size of 40 is that this is the size of our initial space. The respective error decay for $K = 7$ is depicted in Figure 5 (*right*). Note that the good convergence (in particular, similar rates for all outputs of interest simultaneously) is achieved by a rather uneven distribution. The diagrams indicate that, for both values of K, larger eigenvalues as well as possibly double eigenvalues are preferred by the algorithm. This and the fact that, although fewer eigenfunctions are included for the smaller eigenvalues than for the larger ones, but nevertheless the error decay is equal, mean that the eigenfunctions corresponding to larger eigenvalues are effectively used to approximate the ones corresponding to smaller eigenvalues.
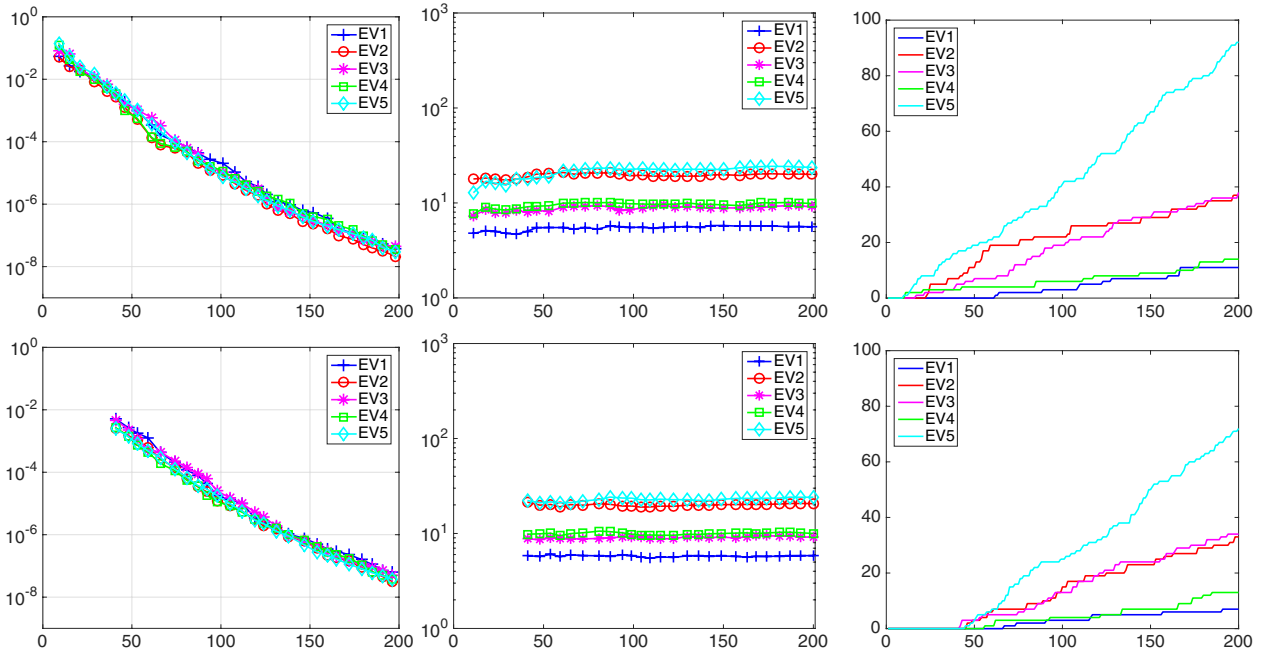
FIGURE 6. Comparison of Algorithm 2 without (*top*) and with (*bottom*) the initialization described in Section 4.3 for $K = 5$ for a selected case in which the greedy algorithm without initialization does not fail. RB error decay (*left*), effectivity numbers (*center*) and accumulated index counts (*right*).

5.4. Effectivity of the greedy method

In this section, we investigate the performance of the greedy method in more detail. For this purpose, we also consider the effectivity numbers γ_i, $1 \leq i \leq K$, of the error estimators and its maximal ratio R defined by

$$\gamma_i := \frac{1}{\#\Xi_{\text{test}}} \sum_{\mu \in \Xi_{\text{test}}} \frac{\eta_i(\mu) \cdot \lambda_i(\mu)}{\lambda_{\text{red},\,i}(\mu) - \lambda_i(\mu)}, \quad R := \frac{\max_{i=1,\dots,K} \gamma_i}{\min_{i=1,\dots,K} \gamma_i}.$$

As already mentioned, the estimators derived in Section 3 are of asymptotic character and therefore generally not reliable for small N. To prevent a misleading selection of basis functions in the first few greedy steps, the initialization described in Section 4.3 is used to generate an initial basis.

Figure 6 shows the error decay (*left*), the effectivity numbers of the *a posteriori* estimators (*center*) and the accumulated index counts (*right*) for $K = 5$ with and without the initialization. In this case, a similar convergence is achieved for both algorithms, and the index count plots also shows a similar behavior. In the preasymptotic range, we observe a difference in the effectivity numbers. Without initialization these numbers possibly depend sensitively on the selected snapshots. While this does not influence the overall performance for $K = 5$, for $K = 7$ we do get extremely poor results if we start directly with the greedy algorithm. This is caused by the fact that the approximation of $\tilde{d}_i(\mu)$ by $d_i(\mu)$ is then not reliable. Thus we always include the initialization step in our adaptive algorithms.

In our experiments, the described initialization always prevents the effectivity numbers from having jumps and leads to good convergence of the greedy methods. For instance, Figure 7 shows the effectivity measures corresponding to the error curves from Figure 4 (*right*) and Figure 5 (*right*). The effectivities are virtually constant and close together which is reflected in a small value of R. This is of crucial importance for the performance
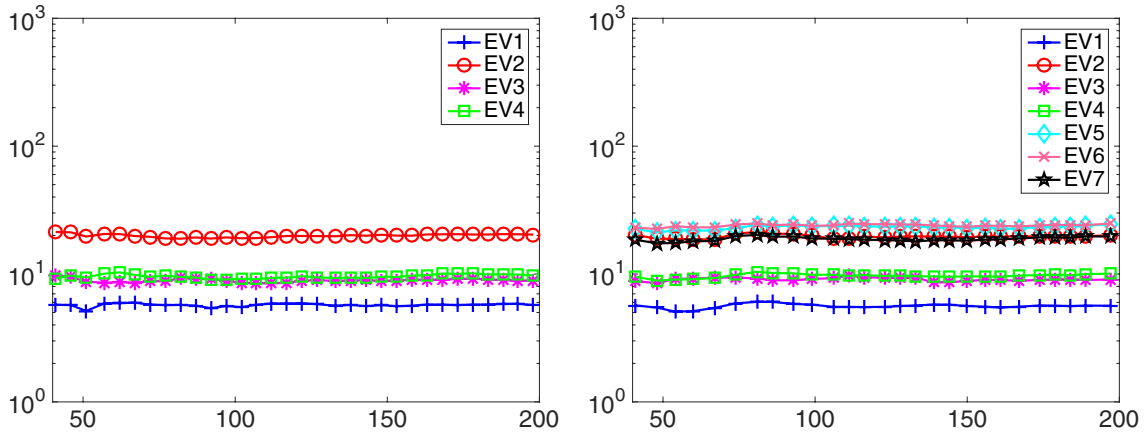
FIGURE 7. Effectivity numbers of the estimators for $K = 4$ (*left*) and $K = 7$ (*right*).
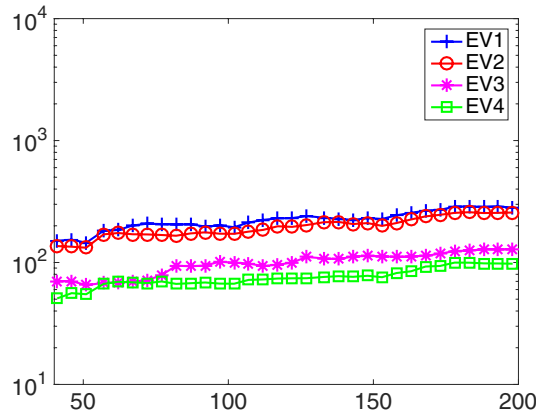

FIGURE 8. Effectivity numbers for the eigenfunctions for $K = 4$.

of our Algorithm 2. In all our settings R is below five, *e.g.*, $R = 3.41$ for $K = 4$. Note that for $K = 4$ and $K = 7$, the same eigenvalues show similar effectivities. A high effectivity ratio R possibly leads to an oversampling of the eigenfunctions associated with the indices of a high effectivity and thus a loss in the performance. At this point although our error estimators are for eigenvalues, we want to show that also the effectivities for the eigenvectors are constant and close together. To do so we depict the results in Figure 8.

After having demonstrated the performance of the single components of our algorithm, let us compare the results of our greedy method using the error estimator and the best components with the convergence of the POD method; *cf.* Section 2.3. Comparing the error plots in Figure 1 with the ones in Figure 4 (*right*) and Figure 5 (*right*), we see that we achieve very similar convergence behavior. In particular, the error curves of our simultaneous reduced basis approximation for the individual eigenvalues are similarly close to each other. Moreover, the accuracy reached at $N \approx 200$ differs only by a factor of roughly ten. We recall that the POD method uses the full training set (namely 10 000 finite element solutions in this case which leads to a computation time of over 10 h) to reach this accuracy while the greedy method only needs a couple of hundred detailed simulations and the evaluation of the estimator which leads in this case to a computation time of 6−7 h.
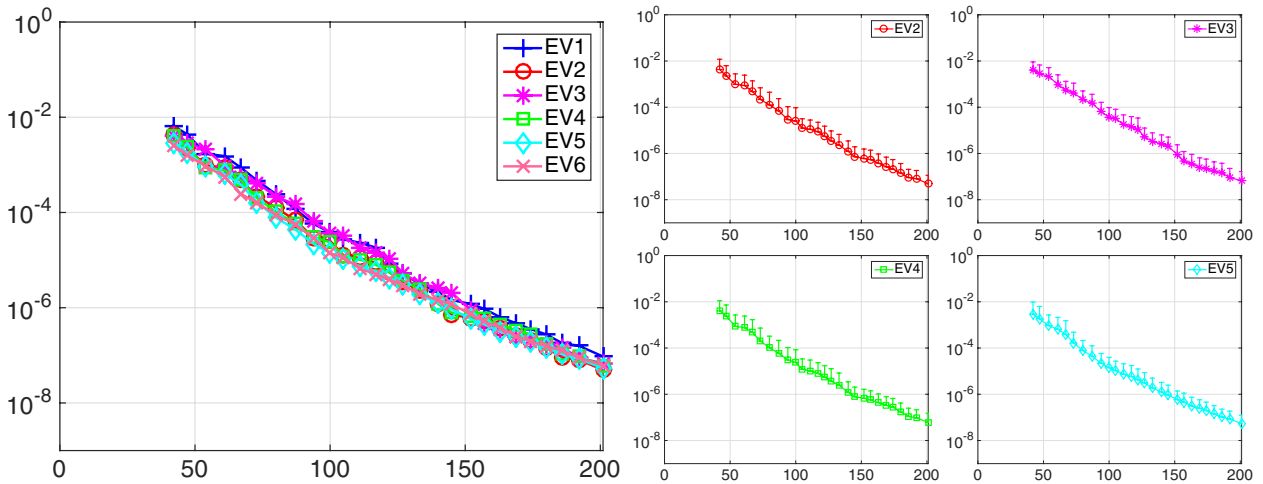
FIGURE 9. RB discretization error for $K = 6$ with standard deviation (as unidirectional error bar) for EV 2 to EV 5.

It should be noted that this gap in computation time between POD and Greedy increases further with the complexity of the detailed solution.

Let us emphasize that the bounds from [32], *i.e.*, $d_i(\mu)^2$ in the denominator of (3.11) instead of $d_i(\mu)$, lead to a large ratio of the maximal and minimal effectivity value and thus to poorer results in the multiple output case. Highly different effectivity numbers result in an over-selection of eigenfunctions associated with the largest effectivity numbers and thus in a performance loss, hence, to a much less attractive greedy algorithm.

Remark 5.1 (Error evaluation). For completeness, Figure 9 shows a convergence plot including the standard deviation for $K = 6$. In the semilogarithmic plots, one can see that the standard deviation is always in the order of the (relative) discretization error itself.

5.5. Speed-up

The speed-up was calculated serially using MATLAB on a Mac laptop; the standard routine eigs, which is based on ARPACK [22], was used for solving the eigenvalue problems. We used linear finite elements for the discretization space V. With our greedy method as introduced above, a significant speed-up in the computation of eigenvalues can be achieved, as is shown for the settings of $K = 2$ to $K = 7$ eigenvalues in Table 1. Here, the calculation of the detailed solutions takes in the range of 3.5 to 3.6 seconds, while the calculation of the reduced solution is possible in 0.021 to 0.078 seconds, resulting in a speed-up of 140 to 43. The higher the value of N, the longer the reconstruction time, but in this case the increase is approximately linear in N.

Moreover, it should be noted that the more accurately the detailed solution is calculated, the more expensive the detailed calculation becomes while the cost for the calculation of the reduced basis solution will stay in the same range, such that we would achieve even higher speed-ups. In computations of practical relevance, the detailed and the reduced accuracies have to be adjusted as it is described in [44]. Here we are mostly interested in the performance of the RB algorithm, and thus we work with a fixed moderate finite element resolution of 15402 DOFs.

As can be seen in Table 2, the computation times for the error estimators (η) as well as for the required offline components for the error estimators, *i.e.*, solution of (3.12) and (3.13) ("Assembly") and computation of $\hat{A}$, increase for increasing values of N. In the case of $\hat{A}$, the increase is approximately linear. Note that these longer computations will only have to be performed in the offline phase and will not have any impact

TABLE 1. Timings for the detailed solution and the online calculations (reduced solution including error estimation; reconstruction of the finite element solution from the reduced solution) in seconds and speed-up numbers.

K	2				4				7			
N	50	100	150	200	50	100	150	200	50	100	150	200
Detailed solution		3.5				3.5				3.6		
Reduced solution	0.025	0.028	–	–	0.037	0.043	0.050	0.055	0.057	0.067	0.075	0.082
Reconstruction	0.009	0.011	–	–	0.009	0.012	0.016	0.019	0.010	0.013	0.018	0.021
Speed-up	140	125	–	–	94	81	70	63	63	53	48	43

TABLE 2. Timings for single components of the offline phase in seconds.

N	η	$\hat{A}$	Assembly
50	0.0028	1.3672	6.076
100	0.0037	2.7640	6.156
150	0.0040	4.1144	6.272
200	0.0046	5.5995	6.312

on the computation times for the online phase. The computation of $g(\mu)$ as defined in (5.1), which is necessary for the error estimator, takes 0.0042 s.

5.6. Wall-slab configuration

In this section, we show the ability of the newly developed reduced basis method to approximate multiple eigenvalues in a two-dimensional wall-slab configuration with a thin elastomer layer in between. The domain shape is an L-shape with three non-overlapping subdomains representing the wall, the elastomer and the slab, denoted by Ω_1, Ω_2 and Ω_3, respectively. The corresponding domains are chosen as $\Omega_1 = [0,1] \times [0,2.8]$, $\Omega_2 = [0,1] \times [2.8,3]$ and $\Omega_3 = [0,3] \times [3,4]$. We again used standard linear finite elements with 30702 DOFs for these calculations.

The material parameters E and ν will again range from $10-100$ and $0.1-0.4$. Since we aim for large numbers of eigenvalues, we perform our simulations for $K = 20$. Figure 10 shows that we do not only obtain very good convergence for the eigenvalues (*left*) but also for the corresponding eigenfunctions (*right*). The error curves chosen to be represented in Figure 10 are representative examples for the eigenvalue and eigenfunction errors in the wall-slab configuration, while the black lines denote the minimum and the maximum of the averaged errors over the $\mu \in \Xi_{\text{test}}$, respectively.

The speed-up is similar to the one analyzed in detail in Section 5.5. For the wall-slab configuration, we show in Table 3 the computation times in the case of $K = 20$ eigenvalues. As can be seen the computation of the detailed solution takes 14.03 seconds, while the computations of the reduced solutions take between 0.14 and 0.24 seconds, depending on the basis size N. This results in a speed-up of 100 for $N = 50$ to 58 if we take $N = 300$ for an accuracy of 10^{-7}.

5.7. Three-dimensional example: First floor building

Since we aim to apply our results to the modal analysis for vibro-accoustics of laminated timber structures, as they occur in modern timber buildings, we test the performance of our method on a three-dimensional geometry representing the first floor of a building. Although wooden structures consist of orthotropic materials, we will use isotropic material parameters for the ease of computation.

Usually different materials are used in the construction of a building. In this case, we have three different materials for the walls. More precisely, we assume that the outer walls are subdomain one, which consists of one
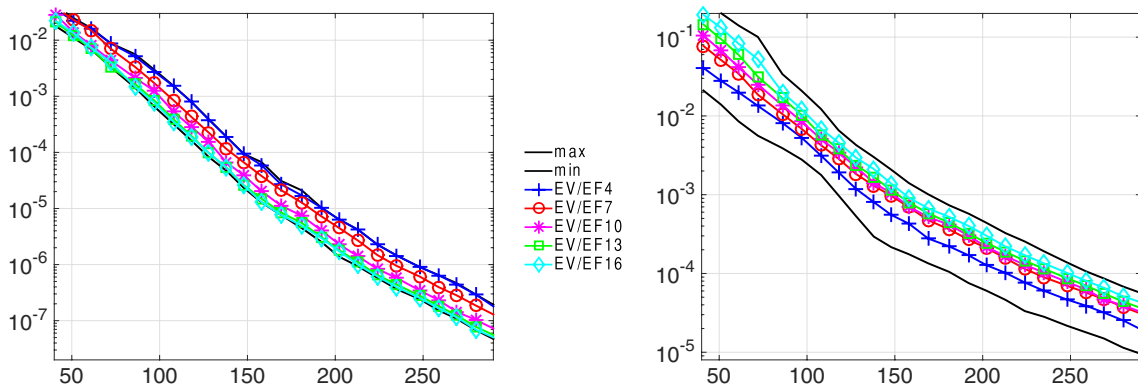
FIGURE 10. Wall-slab configuration with thin elastomer: RB error of eigenvalues (*left*) and eigenfunctions (*right*).

TABLE 3. Timings for the detailed solution and the online calculations (reduced solution including error estimation; reconstruction of the finite element solution from the reduced solution) for a slab-wall configuration in seconds.

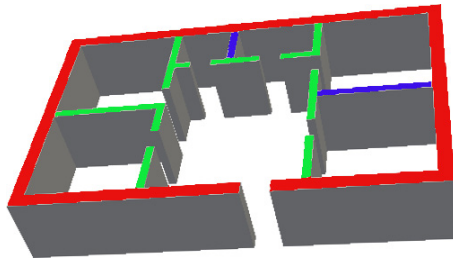| K	20					
N	50	100	150	200	250	300
Detailed solution			14.03			
Reduced solution	0.14	0.16	0.18	0.19	0.22	0.24
Reconstruction	0.027	0.035	0.041	0.049	0.054	0.065
Speed-up	100	87	78	73	63	58

FIGURE 11. Geometry and subdomains: outer walls in red, inner walls load-bearing in green, inner ordinary walls in blue.

material and that the interior walls can be divided into two more subdomains, namely ordinary walls and load-bearing walls. Figure 11 depicts our geometry and the corresponding domains. The material parameters E and ν range from $100-1000$ and $0.1-0.4$. We perform our simulations for $K = 10$ and use standard finite elements with $20\,994$ degrees of freedom.

The first row in Figure 12 represents the first eigenfunctions for three different parameter sets while the second row represents the corresponding fourth eigenfunctions. We used the parameter sets $\mu^1 = (200, 0.1, 800, 0.3, 400, 0.2)$, $\mu^2 = (650, 0.36, 150, 0.25, 900, 0.11)$ and $\mu^3 = (800, 0.3, 500, 0.1, 200, 0.4)$. It can be observed that the eigenfunctions change significantly depending on the parameters while still being approximated very well by our method. Figure 13 shows the error decay for the kth eigenvalues (*left*) and eigenfunctions
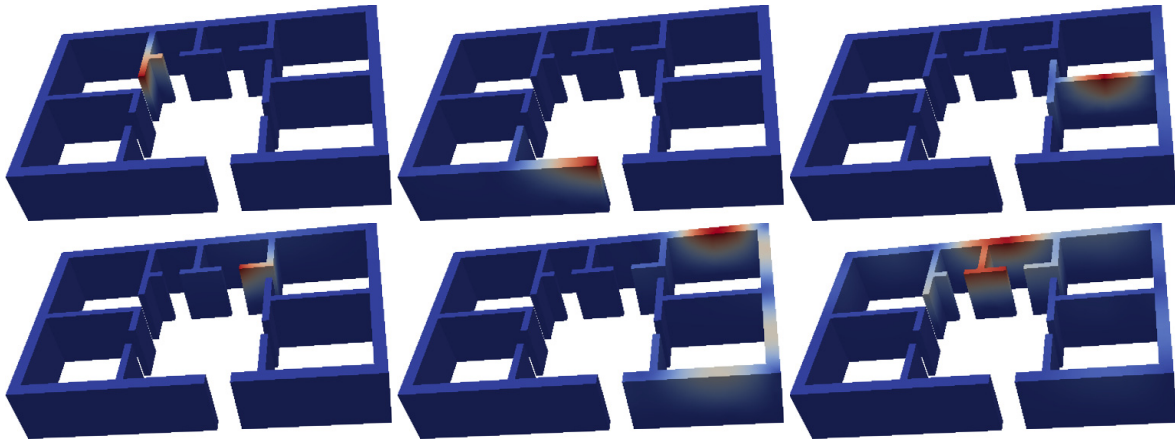
FIGURE 12. Behaviour of the eigenfunctions depending on parameter variations. Top row depicts the first eigenfunction and bottom row the fourth eigenfunction.
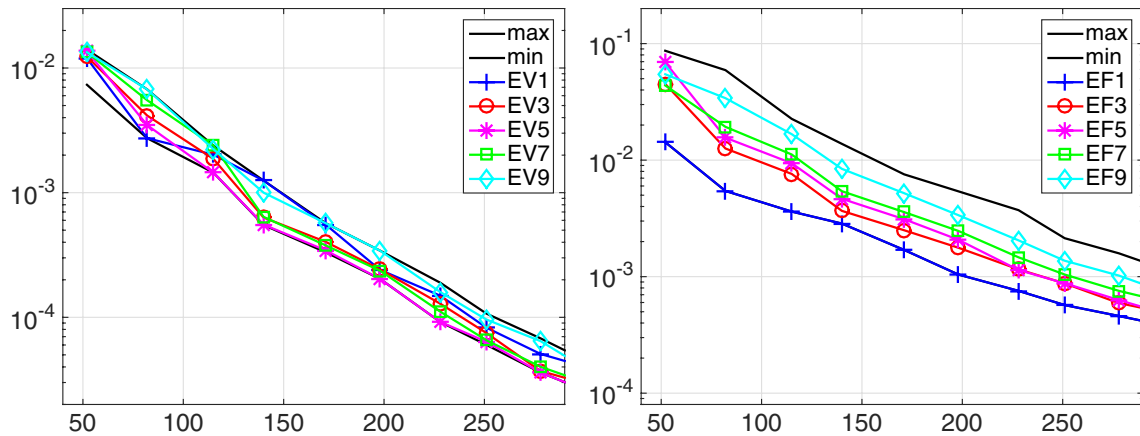

FIGURE 13. First floor configuration: RB error of eigenvalues (*left*) and eigenfunctions (*right*).

TABLE 4. Timings for the detailed solution and the online calculations for the first floor in seconds.

K	10					
N	50	100	150	200	250	300
Detailed solution			31.59			
Reduced solution	0.084	0.096	0.102	0.111	0.125	0.142
Reconstruction	0.021	0.027	0.031	0.035	0.040	0.046
Speed-up	376	329	309	284	252	222

(*right*), $k \in \{1, 3, 5, 7, 9\}$, as well as the minimum and maximum averaged errors. We again obtain very good convergence.

The speed-up in the three-dimensional setting is even more significant. For the first floor of the building, we show in Table 4 the computation times for $K = 10$ eigenvalues. We observe that the computation of the detailed solution takes 31.59 seconds, while the computations of the reduced solutions take between 0.084

and 0.142 seconds, depending on the basis size N. This results in a speed-up of 376 for $N = 50$ to 222 if we take $N = 300$ for an accuracy of 10^{-5}.

6. Conclusion

In this paper, we have developed a model reduction framework for parameterized elliptic eigenvalue problems and applied it numerically to linear elasticity. We have derived an asymptotically reliable error estimator for eigenvalues, even for higher multiplicities, that also facilitates an online-offline decomposition. Several eigenvalues have been approximated simultaneously by a single reduced space for the variational approximation of the eigenvalue problem. Altogether we achieve very effective tailored greedy strategies for the construction of efficient reduced basis spaces for the simultaneous approximation of many eigenvalues.

References

[1] I. Babuška and J. Osborn, Finite Element-Galerkin approximation of the eigenvalues and eigenvectors of selfadjoint problems. *Math. Comput.* **52** (1989) 275–297.

[2] I. Babuška and J. Osborn, Eigenvalue problems, in vol. 2 of *Handbook of Numerical Analysis,* edited by P. Ciarlet and J. Lions. (1991) 641–787.

[3] I. Babuška, B. Guo and J. Osborn, Regularity and numerical solution of eigenvalue problems with piecewise analytic data. *SIAM J. Numer. Anal.* **26** (1989) 1534–1560.

[4] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova and P. Wojtaszczyk, Convergence rates for greedy algorithms in reduced basis methods. *SIAM J. Math. Anal.* **43** (2011) 1457–1472.

[5] 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.

[6] E. Cancès, V. Ehrlacher and T. Lelièvre, Greedy algorithms for high-dimensional eigenvalue problems. *Constructive Approximation* **40** (2014) 387–423.

[7] W. Dahmen, C. Plesken and G. Welper, Double greedy algorithms: Reduced basis methods for transport dominated problems. *ESAIM: M2AN* **48** (2014) 623–663.

[8] R. DeVore, G. Petrova and P. Wojtaszczyk, Greedy algorithms for reduced bases in banach spaces. *Constructive Approximation* **37** (2013) 455–466.

[9] M. Drohmann, B. Haasdonk and M. Ohlberger, Adaptive reduced basis methods for nonlinear convection–diffusion equations. *Finite Volumes for Complex Applications VI Problems & Perspectives* **4** (2011) 369–377.

[10] M. Drohmann, B. Haasdonk, S. Kaulmann and M. Ohlberger, A software framework for reduced basis methods using DUNE-RB and RBmatlab. *Advances in DUNE* (2012) 77–88.

[11] H. Elman and Q. Liao, Reduced basis collocation methods for partial differential equations with random coefficients. *SIAM/ASA J. Uncertainty Quantification* **1** (2013) 192–217.

[12] I. Fumagalli, A. Manzoni, N. Parolini and M. Verani, Reduced basis approximation and *a posteriori* error estimates for parametrized elliptic eigenvalue problems. *ESAIM: M2AN* **50** (2016) 1857–1885.

[13] S. Glas and K. Urban, On non-coercive variational inequalities. *SIAM J. Numer. Anal.* **52** (2014) 2250–2271.

[14] N. Gräbner, S. Quraishi, C. Schröder, V. Mehrmann and U. von Wagner, New numerical methods for the complex eigenvalue analysis of disk brake squeal, in *EuroBrake 2014 Conference Proceedings* (2014).

[15] M. Grepl and A. Patera, *A posteriori* error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *M2AN* **39** (2005) 157–181.

[16] B. Haasdonk and M. Ohlberger, Reduced basis method for finite volume approximations of parametrized linear evolution equations. *M2AN* **42** (2008) 277–302.

[17] B. Haasdonk, J. Salomon and B. Wohlmuth, A reduced basis method for parametrized variational inequalities. *SIAM J. Math. Anal.* **50** (2012) 2656–2676.

[18] T. Horger, S. Kollmannsberger, F. Frischmann, E. Rank and B. Wohlmuth, A new mortar formulation for modeling elastomer bedded structures with modal-analysis in 3D. *Adv. Model. Simul. Eng. Sci.* **2** (2014) 18.

[19] D. Huynh, G. Rozza, S. Sen and A. Patera, A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *C. R. Acad. Sci., Paris, Sér. I* **345** (2007) 473–478.

[20] L. Iapichino, A. Quarteroni, G. Rozza and S. Volkwein, Reduced basis method for the stokes equations in decomposable domains using greedy optimization. *ECMI* **2014** (2014) 1–7.

[21] M. Kahlbacher and S. Volkwein, Galerkin proper orthogonal decomposition methods for parameter dependent elliptic systems. Discussiones Mathematicae: Differential Inclusions, Control and Optimization **27** (2007) 95–117.

[22] R. Lehoucq, D. Sorensen and C. Yang, Arpack users guide: Solution of large-scale eigenvalue problems by implicitely restarted arnoldi methods. *SIAM* (1998).

[23] A. Lovgren, Y. Maday and E. Ronquist, A reduced basis element method for the steady stokes problem. *ESAIM: M2AN* **40** (2006) 529-552.

[24] L. Machiels, Y. Maday, I. Oliveira, A. Patera and D. Rovas, Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *C. R. Acad. Sci., Paris, Sér. I* **331** (2000) 153–158.

[25] Y. Maday, A. Patera and J. Peraire, A general formulation for a posteriori bounds for output functionals of partial differential equations; application to the eigenvalue problem. *C. R. Acad. Sci., Paris, Sér. I* **328** (1999) 823–828.

[26] Y. Maday, A. Patera and G. Turinici, Global a priori convergence theory for reduced basis approximations of single-parameter symmetric coercive elliptic partial differential equations. *C. R. Acad. Sci., Paris, Ser. I* **335** (2002) 289–294.

[27] Y. Maday, A. Patera and G. Turinici, A priori convergence theory for reduced-basis approximations of single-parametric elliptic partial differential equations. *J. Sci. Comput.* **17** (2002) 437–446.

[28] A. Noor and J. Peters, Reduced basis technique for nonlinear analysis of structures. *AIAA J.* **18** (1980) 455–462.

[29] A. Patera and G. Rozza, Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations. Version 1.0, Copyright MIT 2006–2007.

[30] G. Pau, Reduced-basis method for band structure calculations. *Phys. Rev. E* **76** (2007) 046–704.

[31] G. Pau, *Reduced Basis Method for Quantum Models of Crystalline Solids*. Ph. D. thesis, Massachusetts Institute of Technology (2007).

[32] G. Pau, Reduced basis method for simulation of nanodevices. *Phys. Rev. B* **78** (2008) 155–425.

[33] A. Quarteroni, Numerical Models for Differential Problems, Vol. 8 of *MS&A*. Springer, Milan, 2nd edition (2014).

[34] A. Quarteroni, A. Manzoni and F. Negri, *Reduced Basis Methods for Partial Differential Equations. An Introduction*. Springer (2015).

[35] D. Rovas, L. Machiels and Y. Maday, Reduced basis output bound methods for parabolic problems. *IMA J. Numer. Anal.* **26** (2006) 423–445.

[36] G. Rozza and K. Veroy, On the stability of the reduced basis method for stokes equations in parametrized domains. *Comput. Methods. Appl. Mech. Engrg.* **196** (2007) 1244–1260.

[37] G. Rozza, D. Huynh and A. Patera, Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Application to transport and continuum mechanics. *Arch. Comput. Methods Eng.* **15** (2008) 229–275.

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

[39] K. Urban and A.T. Patera, An improved error bound for reduced basis approximation of linear parabolic problems. *Math. Comput.* **83** (2014) 1599–1615.

[40] K. Urban and B. Wieland, Affine decompositions of parametric stochastic processes for application within reduced basis methods, in *7th Vienna International Conference on Mathematical Modelling, MATHMOD 2012, Vienna* (2012) 15–17.

[41] K. Urban, S. Volkwein and O. Zeeb, Greedy sampling using nonlinear optimization, in Vol. 9 of *Reduced Order Methods for Modeling and Computational Reductions* (2014) 137–157.

[42] S. Vallaghe, D.P. Huynh, D.J. Knezevic, T.L. Nguyen and A.T. Patera, Component-based reduced basis for parametrized symmetric eigenproblems. *Adv. Model. Simul. Eng. Sci.* **2** (2015).

[43] K. Veroy, *Reduced-Basis Methods Applied to Problems in Elasticity: Analysis and Applications*. Ph. D. thesis, Massachusetts Institute of Technology (2003).

[44] M. Yano, A minimum-residual mixed reduced basis method: Exact residual certification and simultaneous finite-element reduced-basis refinement. *ESAIM: M2AN* **50** (2016) 163–185.

[45] L. Zanon and K. Veroy-Grepl, The reduced basis method for an elastic buckling problem. *PAMM, Proc. Appl. Math. Mech.* **13** (2013) 439–440.