Chapter 8
Offline Error Bounds for the Reduced Basis Method

Robert O'Connor and Martin Grepl

Abstract The reduced basis method is a model order reduction technique that is specifically designed for parameter-dependent systems. Due to an offline-online computational decomposition, the method is particularly suitable for the many-query or real-time contexts. Furthermore, it provides rigorous and efficiently evaluable a posteriori error bounds, which are used offline in the greedy algorithm to construct the reduced basis spaces and may be used online to certify the accuracy of the reduced basis approximation. Unfortunately, in real-time applications a posteriori error bounds are of limited use. First, if the reduced basis approximation is not accurate enough, it is generally impossible to go back to the offline stage and refine the reduced model; and second, the greedy algorithm guarantees a desired accuracy only over the finite parameter training set and not over all points in the admissible parameter domain. Here, we propose an extension or "add-on" to the standard greedy algorithm that allows us to evaluate bounds over the entire domain, given information for only a finite number of points. Our approach employs sensitivity information at a finite number of points to bound the error and may thus be used to guarantee a certain error tolerance over the entire parameter domain during the offline stage. We focus on an elliptic problem and provide numerical results for a thermal block model problem to validate our approach.

8.1   Introduction

The reduced basis (RB) method is a model order reduction technique that allows efficient and reliable reduced order approximations for a large class of parametrized partial differential equations (PDEs), see e.g. [3, 4, 8, 10, 13, 14] or the review article [12] and the references therein. The reduced basis approximation is build on a so-called "truth" approximation of the PDE, i.e., a usually high-dimensional discretization of the PDE using a classical method such as finite elements or finite

R. O'Connor (✉) • M. Grepl
IGPM, Templergraben 55, Aachen, Germany
e-mail: oconnor@aices.rwth-aachen.de; grepl@igpm.rwth-aachen.de

differences, and the errors in the reduced basis approximation are measured with respect to the truth solution.

The efficiency of the reduced basis method hinges upon an offline-online computational decomposition: In the offline stage the reduced basis spaces are constructed and several necessary precomputations (e.g. projections) are performed. This step requires several solutions of the truth approximation and is thus computationally expensive. In the online stage, given any new parameter value μ in the admissible parameter domain $\mathscr{D}$, the reduced basis approximation and associated a posteriori error bound can be computed very efficiently. The computational complexity depends only on the dimension of the reduced model and not on the dimensionality of the high-dimensional truth space. Due to the offline-online decomposition, the reduced basis method is considered to be beneficial in two scenarios [11, 12]: the many query context, where the offline cost is amortized due to a large number of online solves, and the real-time context, where one simply requires a fast online evaluation.

A crucial ingredient for constructing the reduced basis spaces during the offline stage is the greedy algorithm which was originally proposed in [14]. The greedy algorithm iteratively constructs the reduced space by searching for the largest a posteriori error bound over a finite dimensional parameter train set $\Xi \subset \mathscr{D}$. Once the parameter corresponding to the largest error bound is found, the associated full-order solution is computed, the reduced basis is enriched with this solution, and the necessary quantities for the approximation and error estimation are updated. The process continues until the error bound is sufficiently small, i.e. satisfies a desired error tolerance ϵ_{tol}.

Unfortunately, the desired error tolerance cannot be guaranteed for all parameters in $\mathscr{D}$, but only for all parameters in Ξ. There are usually two arguments to resolve this issue: First, one usually requires the train set Ξ to be chosen "sufficiently" fine, so that a guaranteed certification of Ξ in combination with the smoothness of the solution in parameter space implies a sufficiently accurate reduced basis approximation for all $\mu \in \mathscr{D}$. Second, since the a posteriori error bounds can be efficiently evaluated even in the online stage, one argues that the reduced basis can always be enriched afterwards if a parameter, encountered during the online stage, results in a reduced basis approximation which does not meet the required error tolerance. However, whereas the first argument is heuristic, the second argument—although feasible in the many query context—is not a viable option in the real-time context.

It is this lack of guaranteed offline certification in combination with the real-time context which motivated the development in this paper. Our goal is to develop an approach which allows us to rigorously guarantee a certain accuracy of the reduced basis approximation over the entire parameter domain $\mathscr{D}$, and not just over the train set Ξ. Our method can be considered an "add-on" to the standard greedy algorithm: in addition to the reduced basis approximation and associated a posteriori error bounds we also evaluate the sensitivity information and their error bounds on a finite train set. Given these quantities, we can then bound the error at any

parameter value in the domain and thus bound the accuracy of the reduced basis approximation over $\mathscr{D}$—we use the term "offline bound" for this approach. In that way reduced basis models can be guaranteed to satisfy error tolerances for real-time applications. Obviously, our approach incurs an additional offline cost (see Sect. 8.4) and is thus not useful for applications where one can go back to the offline stage at will and refine the reduced basis approximation at any time. However, if an offline-guaranteed accuracy is essential for the application, the added offline cost may be the only choice and thus acceptable. We note that our results may also be interesting in many-query contexts because they allow us to perform error bounding in the offline stage, reducing the workload in the online stage.

8.2   Problem Statement

For our work it will suffice to directly consider the following truth approximation, i.e., a high-dimensional discretization of an elliptic PDE or just a finite-dimensional system: Given $\mu \in \mathscr{D}$, find $u(\mu) \in X$ such that

$$a(u(\mu), v; \mu) = f(v; \mu), \qquad \forall v \in X. \tag{8.1}$$

Here, $\mathscr{D} \in \mathbb{R}^p$ is a prescribed compact parameter set in which our parameter $\mu = (\mu_1, \ldots, \mu_P)$ resides and X is a suitable (finite-dimensional) Hilbert space with associated inner product $(\cdot, \cdot)_X$ and induced norm $\|\cdot\|_X = \sqrt{(\cdot, \cdot)_X}$. We shall assume that the parameter-dependent bilinear form $a(\cdot, \cdot; \mu) : X \times X \to \mathbb{R}$ is continuous,

$$0 < \gamma(\mu) \equiv \sup_{v \in X} \sup_{w \in X} \frac{a(v, w; \mu)}{\|v\|_X \|w\|_X} \leq \gamma_0 < \infty, \qquad \forall \mu \in \mathscr{D}, \tag{8.2}$$

and coercive,

$$\alpha(\mu) \equiv \inf_{v \in X} \frac{a(v, v; \mu)}{\|v\|_X^2} \geq \alpha_0 > 0, \qquad \forall \mu \in \mathscr{D}, \tag{8.3}$$

and that $f(\cdot; \mu) : X \to \mathbb{R}$ is a parameter-dependent continuous linear functional for all $\mu \in \mathscr{D}$. We shall also assume that (8.1) approximates the real (infinite-dimensional) system sufficiently well for all parameters $\mu \in \mathscr{D}$.

Our assumptions that $a(\cdot, \cdot; \mu)$ be coercive could be relaxed to allow a larger class of operators. The more general class of problems can be handled using the concept of inf-sup stability. For such problems reduced basis methods are well established [14] and our results can easily be adapted.

In addition to the parameter-independent X-norm we also recall the parameter-dependent energy inner product and induced norm $\|\|v\|\|_\mu \equiv \sqrt{a(v, v; \mu)}$. Note that the X-inner product is usually chosen to be equal to the energy inner product for some fixed parameter value $\bar{\mu}$. Generally, sharper error bounds are achieved using the energy-norm. Although we present numerical results for the energy-norm in Sect. 8.6, we will work exclusively with the X-norm in the following derivation to simplify the notation.

We further assume that a and f satisfy the following affine decompositions

$$a(w, v; \mu) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu) a^q(w, v), \quad f(v; \mu) = \sum_{q=1}^{Q_f} \Theta_f^q(\mu) f^q(v), \tag{8.4}$$

where the bilinear forms $a^q(\cdot, \cdot) : X \times X \to \mathbb{R}$ and linear forms $f^q(\cdot) : X \to R$ are independent of the parameters, and the parameter dependent functions $\Theta^q(\cdot) : \mathscr{D} \to \mathbb{R}$ are continuous and are assumed to have derivatives up to a certain order. We also introduce the continuity constants of the parameter independent bilinear and linear forms as

$$\gamma_{a,q} \equiv \sup_{v \in X} \sup_{w \in X} \frac{a^q(v, w)}{\|v\|_X \|w\|_X} \quad \text{and} \quad \gamma_{f,q} \equiv \sup_{v \in X} \frac{f^q(v)}{\|v\|_X}. \tag{8.5}$$

8.2.1 Sensitivity Derivatives

In order to understand how solutions of (8.1) behave in the neighborhood of a given parameter value μ we consider sensitivity derivatives. Given a parameter $\mu \in \mathscr{D}$ and associated solution $u(\mu)$ of (8.1), the directional derivative $\nabla_\eta u(\mu) \in X$ in the direction $\eta \in \mathbb{R}^p$ is given as the solution to

$$a(\nabla_\eta u(\mu), v; \mu) = \sum_{q=1}^{Q_f} \left[\nabla_\eta \Theta_f^q(\mu)\right] f^q(v) - \sum_{q=1}^{Q_a} \left[\nabla_\eta \Theta_a^q(\mu)\right] a^q(u(\mu), v), \ \forall v \in X. \tag{8.6}$$

Often, we will need to solve for $u(\mu)$ and several of its sensitivity derivatives. In that case we can take advantage of the fact that both (8.1) and (8.6) have the same η-independent operator on the left-hand side.

8.3   The Reduced Basis Method

8.3.1   Approximation

The reduced basis method involves the Galerkin projection of the truth system onto a much lower-dimensional subspace X_N of the truth space X. The space X_N is spanned by solutions of (8.1), i.e., $X_N = \text{span}\{u(\eta^n),\ 1 \le n \le N\}$, where the parameter values η^n are selected by the greedy algorithm [14].

The reduced basis approximation of (8.1) is thus: Given $\mu \in \mathscr{D}$, $u_N(\mu) \in X_N$ satisfies

$$a(u_N(\mu), v; \mu) = f(v), \qquad \forall v \in X_N. \tag{8.7}$$

The definition of the sensitivity derivatives $\nabla_\eta u_N$ is analogous to (8.6) and thus omitted. We also note that—given the assumptions above—the reduced basis approximation $u_N(\mu)$ can be efficiently computed using the standard offline-online decomposition.

8.3.2   A Posteriori Error Estimation

In the sequel we require the usual a posteriori bounds for the error $e(\mu) \equiv u(\mu) - u_N(\mu)$ and for its sensitivity derivatives. To this end, we introduce the residual associated with (8.7) and given by

$$r(v; \mu) = f(v; \mu) - a(u_N(\mu), v; \mu), \qquad \forall v \in X, \tag{8.8}$$

as well as the residual associated with the sensitivity equation and given by

$$r_\eta(v; \mu) \equiv \sum_{q=1}^{Q_f} \nabla_\eta \Theta_f^q(\mu) f^q(v) - a\left(\nabla_\eta u_N(\mu), v; \mu\right)$$

$$- \sum_{q=1}^{Q_a} \nabla_\eta \Theta_a^q(\mu) a^q(u_N(\mu), v), \qquad \forall v \in X. \tag{8.9}$$

We also require a lower bound for the coercivity constant, $\alpha_{\text{LB}}(\mu)$, satisfying $0 < \alpha_0 \le \alpha_{\text{LB}}(\mu) \le \alpha(\mu),\ \forall \mu \in \mathscr{D}$; the calculation of such lower bounds is discussed in Sect. 8.5.

We next recall the well known a posteriori bounds for the error in the reduced basis approximation and its sensitivity derivative; see e.g. [12] and [9] for the proofs.

Theorem 1 *The error in the reduced basis approximation, $e(\mu) = u(\mu) - u_N(\mu)$, and its sensitivity derivative, $\nabla_\eta e(\mu) = \nabla_\eta u(\mu) - \nabla_\eta u_N(\mu)$, are bounded by*

$$\|e(\mu)\|_X \le \Delta(\mu) \equiv \frac{\|r(\cdot;\mu)\|_{X'}}{\alpha_{\mathrm{LB}}(\mu)}. \tag{8.10}$$

and

$$\|\nabla_\eta e(\mu)\|_X \le \Delta_\eta(\mu) \equiv \frac{1}{\alpha_{\mathrm{LB}}(\mu)} \left(\|r_\eta(\cdot;\mu)\|_{X'} + \Delta(\mu) \sum_{q=1}^{Q_a} |\nabla_\eta \Theta_a^q(\mu)| \gamma_{a,q} \right). \tag{8.11}$$

The bounds given in (8.10) and (8.11)—similar to the approximations $u(\mu)$ and $\nabla_\eta u_N(\mu)$—can all be computed very cheaply in the online stage; see [12] for details.

8.4 Offline Error Bounds

Our goal in this section is to derive error bounds which can be evaluated efficiently at any parameter value ω in a specific domain while only requiring the solution of the RB model at one fixed parameter value (i.e. anchor point) μ. Obviously, such bounds will be increasingly pessimistic as we deviate from the anchor point μ and will thus only be useful in a small neighborhood of μ. However, such bounds can be evaluated offline and thus serve as an "add-on" to the greedy procedure in order to guarantee a "worst case" accuracy over the whole parameter domain.

8.4.1 Bounding the Difference Between Solutions

As a first ingredient we require a bound for the differences between solutions to (8.1) at two parameter values μ and ω. We note that the analogous bounds stated here for the truth solutions will also hold for solutions to the reduced basis model.

Theorem 2 *The difference between two solutions, $d(\mu, \omega) \equiv u(\mu) - u(\omega)$, satisfies*

$$\|d(\mu, \omega)\|_X \le \frac{1}{\alpha_{\mathrm{LB}}(\omega)} \left(\|u\|_X \sum_{q=1}^{Q_a} \gamma_{a,q} |\Theta_a^q(\mu) - \Theta_a^q(\omega)| \right.$$
$$\left. + \sum_{q=1}^{Q_f} \gamma_{f,q} |\Theta_f^q(\mu) - \Theta_f^q(\omega)| \right). \tag{8.12}$$

Proof We first take the difference of two solutions of (8.1) for μ and ω, add $\pm \sum_{q=1}^{Q_a} \Theta_a^q(\omega) a^q(u(\mu), v)$, and invoke (8.4) to arrive at

$$a(d(\mu, \omega), v; \omega) = \sum_{q=1}^{Q_f} \left(\Theta_f^q(\mu) - \Theta_f^q(\omega) \right) f^q(v)$$

$$- \sum_{q=1}^{Q_a} \left(\Theta_a^q(\mu) - \Theta_a^q(\omega) \right) a^q(u(\mu), v). \qquad (8.13)$$

Following the normal procedure we choose $v = d(\mu, \omega)$ which allows us to bound the left-hand side using (8.3) and the coercivity lower bound. On the right-hand side we make use of the triangle inequality and invoke (8.5) to obtain

$$\alpha_{\text{LB}}(\omega) \|d(\mu, \omega)\|_X^2 \leq \|d(\mu, \omega)\|_X \left(\sum_{q=1}^{Q_f} \gamma_{f,q} \left| \Theta_f^q(\mu) - \Theta_f^q(\omega) \right| \right.$$

$$\left. + \|u(\mu)\|_X \sum_{q=1}^{Q_a} \gamma_{a,q} \left| \Theta_a^q(\mu) - \Theta_a^q(\omega) \right| \right). \qquad (8.14)$$

Cancelling and rearranging terms gives the desired result. □

Similarly, we can bound the difference between the sensitivity derivatives at various parameter values as stated in the following theorem. The proof is similar to the proof of Theorem 2 and thus omitted.

Theorem 3 *The difference between $\nabla_\eta u(\mu)$ and $\nabla_\eta u(\omega)$ satisfies the following bounding property*

$$\|\nabla_\eta d(\mu, \omega)\|_X \leq \sum_{q=1}^{Q_f} \frac{\gamma_{f,q}}{\alpha_{\text{LB}}(\omega)} |\nabla_\eta \Theta_f^q(\mu) - \nabla_\eta \Theta_f^q(\omega)|$$

$$+ \sum_{q=1}^{Q_a} \frac{\gamma_{a,q}}{\alpha_{\text{LB}}(\omega)} \left(|\Theta_a^q(\mu) - \Theta_a^q(\omega)| \|\nabla_\eta u(\mu)\|_X + |\nabla_\eta \Theta_a^q(\omega)| \|d(\mu, \omega)\|_X \right.$$

$$\left. + |\nabla_\eta \Theta_a^q(\mu) - \nabla_\eta \Theta_a^q(\omega)| \|u(\mu)\|_X \right). \qquad (8.15)$$

We make two remarks. First, we again note that Theorems 2 and 3 also hold for the reduced basis system when all quantities are changed to reduced basis quantities. Second, in the sequel we also require the bounds (8.12) and (8.15). Unfortunately, these cannot be computed online-efficiently since they involve the truth quantities

$\|u(\mu)\|_X$ and $\|\nabla_\eta u(\mu)\|_X$. However, we can invoke the triangle inequality to bound e.g. $\|u(\mu)\|_X \leq \|u_N(\mu)\|_X + \Delta(\mu)$ and similarly for $\|\nabla_\eta u(\mu)\|_X$. We thus obtain efficiently evaluable upper bounds for (8.12) and (8.15).

8.4.2 An Initial Offline Bound

We first consider error bounds that do not require the calculation of sensitivity derivatives. To this end we assume that the reduced basis approximation (8.7) has been solved and that the bound (8.10) has been evaluated for the parameter value $\mu \in \mathscr{D}$. We would then like to bound $e(\omega) = u(\omega) - u_N(\omega)$ for all $\omega \in \mathscr{D}$. This bound, as should be expected, will be useful only if ω is sufficiently close to μ.

Theorem 4 *Given a reduced basis solution $u_N(\mu)$ and associated error bound $\Delta(\mu)$ at a specific parameter value μ, the reduced basis error at any parameter value $\omega \in \mathscr{D}$ is bounded by*

$$\|e(\omega)\|_X \leq \Delta^0(\mu, \omega) \equiv \Delta(\mu) + \frac{2}{\alpha_{\mathrm{LB}}(\omega)} \left(\sum_{q=1}^{Q_f} \gamma_{f,q} \left| \Theta_f^q(\mu) - \Theta_f^q(\omega) \right| \right)$$

$$+ \frac{2\|u_N(\mu)\|_X + \Delta(\mu)}{\alpha_{\mathrm{LB}}(\omega)} \left(\sum_{q=1}^{Q_a} \gamma_{a,q} \left| \Theta_a^q(\mu) - \Theta_a^q(\omega) \right| \right). \qquad (8.16)$$

Proof We begin by writing $e(\omega)$ in terms of $e(\mu)$, i.e.

$$e(\omega) = e(\mu) - d(\mu, \omega) + d_N(\mu, \omega). \qquad (8.17)$$

We then take the X-norm of both sides and apply the triangle inequality to the right-hand side. Invoking (8.10) and (8.12) gives the desired result. □

We again note that we only require the reduced basis solution and the associated a posteriori error bound at the parameter value μ to evaluate the a posteriori error bound proposed in (8.16). Furthermore, the bound reduces to the standard bound $\Delta(\mu)$ for $\omega = \mu$, but may increase rapidly as ω deviates from μ. To alleviate this issue, we propose a bound in the next section that makes use of first-order sensitivity derivatives.

8.4.3 Bounds Based on First-Order Sensitivity Derivatives

We first note that we can bound the error in the sensitivity derivative $\nabla_\eta e$ at the parameter value ω as follows.

Theorem 5 *The error in the reduced basis approximation of the sensitivity derivative at any parameter value ω satisfies*

$$\|\nabla_\eta e(\omega)\|_X \leq \Delta_\eta(\mu) + \|\nabla_\eta d(\mu, \omega)\|_X + \|\nabla_\eta d_N(\mu, \omega)\|_X. \tag{8.18}$$

Proof The result directly follows from $\nabla_\eta e(\omega) = \nabla_\eta u(\omega) - \nabla_\eta u_N(\omega)$ by adding and subtracting $\pm\nabla_\eta u(\mu)$ and $\pm\nabla_\eta u_N(\mu)$, rearranging terms, and invoking the triangle inequality. $\square$

Given the previously derived bounds for the sensitivity derivatives, we can now introduce an improved bound in the following theorem.

Theorem 6 *Making use of sensitivity derivatives we get the following error bound for the parameter value $\omega = \mu + \rho \in \mathcal{D}$:*

$$\|e(\omega)\|_X \leq \Delta^1(\mu, \omega) \equiv \Delta(\mu) + \Delta_\rho(\mu) + \frac{2}{\alpha_{LB}} \sum_{q=1}^{Q_f} \gamma_{f,q} I_{f,\nabla}^q$$

$$+ \sum_{q=1}^{Q_a} \frac{\gamma_{a,q}}{\alpha_{LB}} \left(\left(2\|u_N(\mu)\|_X + \Delta(\mu)\right) \left(I_{a,\nabla}^q + \sum_{\bar{q}=1}^{Q_a} \frac{\gamma_{a,\bar{q}} I_{a,a}^{q,\bar{q}}}{\alpha_{LB}} \right) \right.$$

$$\left. + \left(2\|\nabla_\rho u_N(\mu)\|_X + \Delta_\rho(\mu)\right) I_a^q + \frac{2}{\alpha_{LB}} \sum_{\bar{q}=1}^{Q_f} \gamma_{f,\bar{q}} I_{a,f}^{q,\bar{q}} \right), \tag{8.19}$$

where the coercivity lower bound α_{LB} satisfies $\alpha_{LB} \leq \min_{0 \leq \tau \leq 1} \alpha(\mu + \tau\rho)$ and the integrals are given by

$$I_{f,\nabla}^q \equiv \int_0^1 \left| \nabla_\rho \Theta_f^q(\mu) - \nabla_\rho \Theta_f^q(\mu + \tau\rho) \right| d\tau, \tag{8.20a}$$

$$I_{a,\nabla}^q \equiv \int_0^1 \left| \nabla_\rho \Theta_a^q(\mu) - \nabla_\rho \Theta_a^q(\mu + \tau\rho) \right| d\tau, \tag{8.20b}$$

$$I_a^q \equiv \int_0^1 \left| \Theta_a^q(\mu) - \Theta_a^q(\mu + \tau\rho) \right| d\tau, \tag{8.20c}$$

$$I_{a,f}^{q,\bar{q}} \equiv \int_0^1 \left| \nabla_\rho \Theta_a^q(\mu + \tau\rho) \right| \left| \Theta_f^{\bar{q}}(\mu) - \Theta_f^{\bar{q}}(\mu + \tau\rho) \right| d\tau, \tag{8.20d}$$

$$I_{a,a}^{q,\bar{q}} \equiv \int_0^1 \left| \nabla_\rho \Theta_a^q(\mu + \tau\rho) \right| \left| \Theta_a^{\bar{q}}(\mu) - \Theta_a^{\bar{q}}(\mu + \tau\rho) \right| d\tau. \tag{8.20e}$$

Proof We begin with the fundamental theorem of calculus

$$e(\mu + \rho) = e(\mu) + \int_0^1 \nabla_\rho e(\mu + \tau\rho) \, d\tau. \tag{8.21}$$

We then take the X-norm of both sides and apply the triangle inequality on the right-hand side. Invoking Theorems 1, 3, and 5 leads to the desired result.   □

In the majority of cases the functions $\Theta_a^q(\cdot)$ and $\Theta_f^q(\cdot)$ are analytical functions, and the integrals in (8.20) can be evaluated exactly. Nevertheless, we only really need to bound the integrals uniformly over certain neighborhoods [2].

The bounds given in Theorems 4 and 6 allow us to bound the error anywhere in the neighborhood of a parameter value μ using only a finite number of reduced basis evaluations, i.e., the reduced basis approximation and the sensitivity derivative as well as their a posteriori error bounds. In practice, we first introduce a tessellation of the parameter domain $\mathscr{D}$ with a finite set of non-overlapping patches. We then perform the reduced basis calculations at one point (e.g. the center) in each patch and evaluate the offline error bounds (8.16) or (8.19) over the rest of each patch. Figure 8.1 shows a sketch of the typical behaviour of the offline bounds for a one-dimensional parameter domain. For a given fixed training set of size n_{train}, the additional cost to evaluate the first-order bounds during the offline stage is at most P times higher than the "classical" greedy search for a P dimensional parameter (only considering the greedy search and not the computation of the basis functions). This can be seen from Theorem 6, i.e., in addition to evaluating the RB approximation and error bound at all n_{train} parameter values, we also need to evaluate the sensitivity derivative and the respective error bound at these parameter values.

We note, however, that the local shape of the offline bounds as shown in Fig. 8.1 might not be of interest. Instead, we are usually interested in the global shape and in the local worst case values which occur at the boundaries of the patches, i.e. the

Fig. 8.1 Results using the first-order offline bounds
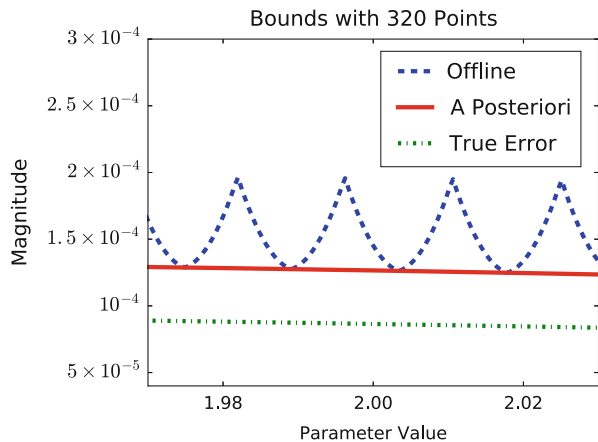
peaks of the blue dashed line in Fig. 8.1. In the numerical results presented below we therefore only plot the upper bound obtained by connecting these peaks.

There is just one ingredient that is still missing: calculating lower bounds for the coercivity constants.

8.5 Computing Coercivity Constants

In reduced basis modeling stability constants play a vital role, but finding efficient methods to produce the lower bounds that we need is notoriously difficult. For simple problems tricks may exist to evaluate such lower bounds exactly [6], but for the majority of problems more complicated methods are needed. The most used method seems to be the successive constraints method (SCM). It is a powerful tool for calculating lower bounds for coercivity constants at a large number of parameter values while incurring minimal cost [1, 5].

Let us introduce the set

$$\mathcal{Y} \equiv \{y \in \mathbb{R}^Q | y_q = a^q(v, v)/\|v\|_X^2, \forall 1 \le 1 \le Q \text{ and any } v \in X\}. \tag{8.22}$$

The coercivity constant can be written as the solution to an optimization problem over $\mathcal{Y}$.

$$\alpha(\mu) = \inf_{y \in \mathcal{Y}} \sum_{q=1}^Q \Theta_a^q(\mu) y_q \tag{8.23}$$

Working with this formulation of the coercivity constant is often easier than working with (8.3). The main difficulty is that the set $\mathcal{Y}$ is only defined implicitly and can be very complicated. The idea of SCM is to relax the optimization problem by replacing $\mathcal{Y}$ with a larger set that is defined by a finite set of linear constraints. Lower bounds for the coercivity constant are then given implicitly as the solution to a linear programming problem.

Unfortunately, SCM will not suffice for our purposes. We will need explicit bounds on the coercivity constant over regions of the parameter domain. It was shown how such bounds can be obtained in a recent paper [7]. That paper makes use of SCM and the fact that $\alpha(\mu)$ is a concave function of the variables $\Theta_a^q(\mu)$. The concavity can be shown from (8.23) and tells us that lower bounds can be derived using linear interpolation.

8.6   Numerical Results

We consider the standard thermal block problem [12] to test our approach. The spatial domain, given by $\Omega = (0, 1)^2$ with boundary Γ, is divided into four equal squares denoted by Ω_i, $i = 1, \ldots, 4$. The reference conductivity in Ω_0 is set to unity, we denote the normalized conductivities in the other subdomains Ω_i by σ_i. The conductivities will serve as our parameters and vary in the range $[0.5, 5]$. We consider two problem settings: a one parameter and a three parameter problem; the domains of our test problems are shown in Fig. 8.2. The temperature satisfies the Laplace equation in Ω with continuity of temperature and heat flux across subdomain interfaces. We assume homogeneous Dirichlet boundary conditions on the bottom of the domain, homogeneous Neumann on the left and right side, and a unit heat flux on the top boundary of the domain. The weak formulation of the problem is thus given by (8.1), with the bilinear and linear forms satisfying the assumptions stated in Sect. 8.2. The derivation is standard and thus omitted. Finally, we introduce a linear truth finite element subspace of dimension $\mathcal{N} = 41,820$. We also define the X-norm to be equal to the energy-norm with $\sigma_i = 1$ for all $i \in \{1, \ldots, 4\}$.

For this example problem our bounds can be greatly simplified. The most obvious simplification is that all terms involving $\Theta_f^q(\cdot)$ can be eliminated due to the fact that $f(\cdot; \mu)$ is parameter independent. We also not that the $\Theta_a^q(\cdot)$ are affine and that their derivatives are constant. As a result the integrals given in (8.20a), (8.20b), and (8.20d) are all equal to zero and the evaluation of (8.20c) and (8.20e) is trivial.

For our first example problem we set $\sigma_0 = \sigma_1 = \sigma_3 = 1$ and thus have one parameter $\mu = \sigma_2 \in [0.5, 5]$. We build a four-dimensional reduced basis model with X_N spanned by solutions $u(\zeta)$ at $\zeta \in \{0.6, 1.35, 2.75, 4\}$. The offline bounds are calculated by dividing the parameter domain into ℓ uniform intervals in the log scale, computing the reduced basis quantities at the center of each interval, and computing offline bounds for the rest of each interval. Figure 8.3 shows the a posteriori error bounds and the zeroth-order offline bounds for $\ell \in \{320, 640, 1280\}$. Here the detailed offline error bounds are not plotted but rather curves that interpolate the peaks of those bounds. We note that the actual offline bounds lie between the plotted curves and the a posteriori bounds and vary very quickly between ℓ valleys and $\ell + 1$

Fig. 8.2 2×2 thermal block model problem. (**a**) Thermal block with 1 parameter. (**b**) Thermal block with 3 parameter
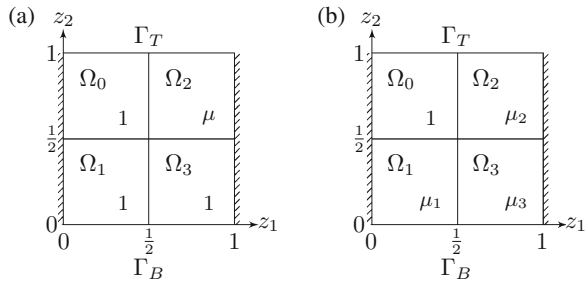
Fig. 8.3 Results using the zero-order offline bounds
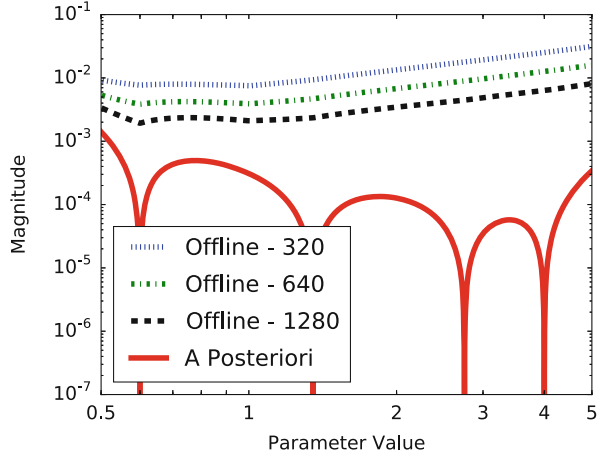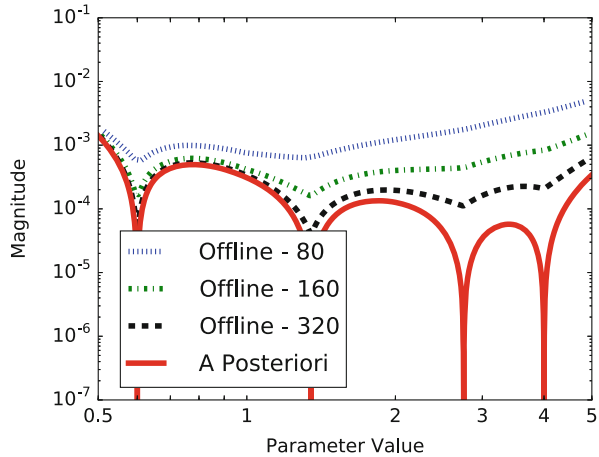

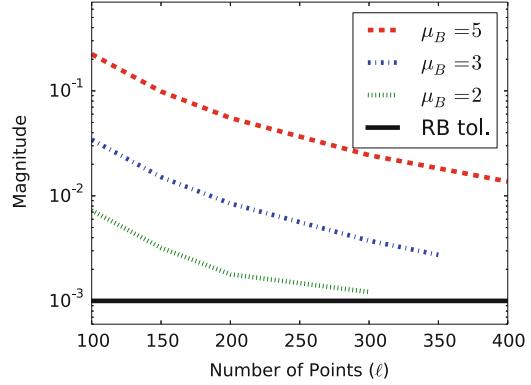**Fig. 8.4** Results using the first-order offline bounds

peaks. For all practical applications the quick variations are unimportant and only the peaks are of interest, since they represent the worst case upper bound.

We observe that in comparison with the a posteriori bounds the zeroth-order bounds are very pessimistic. We can achieve much better results, i.e. tighter upper bounds, by using the first-order bounds as shown in Fig. 8.4. The first-order bounds are much smaller, although reduced basis computations were performed at fewer points in the parameter domain. Figure 8.1 shows the detailed behavior of the offline bounds with $\ell = 320$ over a small part of the parameter domain.

Depending on the tolerance ϵ_{tol} that we would like to satisfy, a uniform log scale distribution of the ℓ points will not be optimal. In practice, it may be more effective to add points adaptively wherever the bounds need to be improved.

We next consider the three parameter problem setting with the admissible parameter domain $\mathscr{D} = [0.5, \mu_B]^3$, where μ_B is the maximal admissible parameter value. This time we divide each interval $[0.5, \mu_B]$ into ℓ log-scale uniform subintervals and

Fig. 8.5 Offline bounds for
the 3D problem with various
parameter domains

take the tensor product to get ℓ^3 patches in $\mathscr{D}$. We then compute the offline bounds
over these patches. For this problem we only use the first-order offline bounds
because using the zeroth-order bounds would be too expensive. Figure 8.5 shows
the maximum values of the offline error bounds over the entire domain for various
values of ℓ and three different values of μ_B. We observe that a larger parameter
range of course requires more anchor points to guarantee a certain desired accuracy,
but also that the accuracy improves with the number of anchor points.

8.7 Conclusions and Extensions

The main result of this work is the derivation of error bounds which can be computed
offline and used to guarantee a certain desired error tolerance over the whole
parameter domain. This allows us to shift the cost of evaluating error bounds to
the offline stage thus reducing the online computational cost, but more importantly it
allows us to achieve a much higher level of confidence in our models. It enables us to
apply reduced basis methods to real-time applications while ensuring the accuracy
of the results.

It should be noted that our methods produce pessimistic bounds and can be quite
costly. Furthermore, since the bounds are based on sensitivity information—similar
to the approach presented in [2]—the approach is restricted to a modest number of
parameters. In general the heuristic method may be more practical unless it is really
necessary to be certain that desired tolerances are met.

We have derived zero and first-order offline bounds. We expect that using higher-
order bounds would produce better results and reduce the computational cost.
It may also be interesting to tailor the reduced basis space to produce accurate
approximations of not only the solution but also of its derivatives. Furthermore,
the proposed bounds may also be used to adaptively refine the train set, i.e. we
start with a coarse train set and then adaptively refine the set parameter regions

where the offline bounds are largest (over the tessellations/patches). This idea will be investigated in future research.

In practical applications it will usually be more useful to deal with outputs rather than the full state $u(\mu)$. The reduced basis theory for such problems is well established [12], and the results that we present here can easily be adapted.

Many of the ideas and bounds given in this paper could also be used to optimize reduced basis models. One could for example attempt to optimize the train samples that are used in greedy algorithms. If using offline bounds is too costly, the theory can still be useful to derive better heuristics for dealing with error tolerances.

One example of real-time problems where offline bounds could be used is adaptive parameter estimation. In such contexts the system's parameters are unknown meaning that we cannot use a posteriori bounds. We can, however, use offline bounds.

References

1. Chen, Y., Hesthaven, J.S., Maday, Y., Rodríguez, J.: Improved successive constraint method based a posteriori error estimate for reduced basis approximation of 2D Maxwell's problem. ESAIM: Math. Model. Numer. Anal. **43**(6), 1099–1116 (2009)
2. Eftang, J.L., Grepl, M.A., Patera, A.T.: A posteriori error bounds for the empirical interpolation method. C. R. Math. **348**, 575–579 (2010)
3. Grepl, M.A., Patera, A.T.: A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. ESAIM: Math. Model. Numer. Anal. **39**(1), 157–181 (2005)
4. Haasdonk, B., Ohlberger, M.: Reduced basis method for finite volume approximations of parametrized linear evolution equations. ESAIM: Math. Model. Numer. Anal. **42**(02), 277–302 (2008)
5. Huynh, D.B.P., Rozza, G., Sen, S., Patera, A.T.: A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. C. R. Acad. Sci. Paris, Ser. I **345**(8), 473–478 (2007)
6. Nguyen, N.C., Veroy, K., Patera, A.T.: Certified real-time solution of parametrized partial differential equations. In: Yip, S. (ed.) Handbook of Materials Modeling, chap. 4.15, pp. 1523–1558. Springer, New York (2005)
7. O'Connor, R.: Bounding stability constants for affinely parameter-dependent operators. C. R. Acad. Sci. Paris, Ser. I **354**(12), 1236–1240 (2016)
8. O'Connor, R.: Lyapunov-based error bounds for the reduced-basis method. IFAC-PapersOnLine **49**(8), 1–6 (2016)
9. Oliveira, I., Patera, A.T.: Reduced-basis techniques for rapid reliable optimization of systems described by affinely parametrized coercive elliptic partial differential equations. Optim. Eng. **8**(1), 43–65 (2007)
10. Prud'homme, C., Rovas, D.V., Veroy, K., Machiels, L., Maday, Y., Patera, A.T., Turinici, G.: Reliable real-time solution of parametrized partial differential equations: reduced-basis output bound methods. J. Fluid. Eng. **124**(1), 70–80 (2002)
11. Quarteroni, A., Rozza, G., Manzoni, A.: Certified reduced basis approximation for parametrized partial differential equations and applications. J. Math. Ind. **1**(3), 1–49 (2011)
12. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Arch. Comput. Meth. Eng. **15**(3), 229–275 (2008)

13. Veroy, K., Prud'homme, C., Patera, A.T.: Reduced-basis approximation of the viscous burgers equation: rigorous a posteriori error bounds. C.R. Math. **337**(9), 619–624 (2003)
14. Veroy, K., Prud'homme, C., Rovas, D.V., Patera, A.T.: A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In: Proceedings of the 16th AIAA Computational Fluid Dynamics Conference (2003). AIAA Paper 2003–3847