

Jens Lohne Eftang

Reduced basis methods for parametrized partial differential equations

Thesis for the degree of Philosophiae Doctor

Trondheim, May 2011

Norwegian University of Science and Technology
Faculty of Information Technology, Mathematics
and Electrical Engineering
Department of Mathematical Sciences



NTNU – Trondheim
Norwegian University of
Science and Technology

NTNU

Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Information Technology, Mathematics
and Electrical Engineering
Department of Mathematical Sciences

© Jens Lohne Eftang

ISBN 978-82-471-2821-3 (printed ver.)
ISBN 978-82-471-2822-0 (electronic ver.)
ISSN 1503-8181

Doctoral theses at NTNU, 2011:140

Printed by NTNU-trykk

PREFACE

This PhD thesis concludes my doctoral studies in numerical analysis at the Department of Mathematical Sciences at the Norwegian University of Science and Technology (NTNU).

I am indebted to my thesis advisor, Professor Einar M. Rønquist. I wish to thank him for his encouragement, good advice, and insight, and for introducing me to the field of reduced basis methods.

In 2009 and 2010, I had the pleasure of visiting Professor Anthony T. Patera at Massachusetts Institute of Technology (MIT) for a full academic year. I wish to thank Professor Patera for all his help and advice, and in particular for making me feel like a full member of his research group both during and after my stay. Thanks also to Ms Debra Blanchard for making my stay at MIT possible. I'm grateful to Dr. David Knezevic and to Dr. Phuong Huynh in Professor Patera's group for extensive collaboration, fruitful discussion, and in particular co-authorship on some of the papers included in this thesis.

I am also grateful to Professor Martin A. Grepl, RWTH Aachen, and to Professor Benjamin Stamm, University of California, Berkeley, for collaboration, good discussion, and co-authorship.

I wish to thank my friends at NTNU for the always enjoyable and sometimes endless coffee breaks during work hours and for an occasional beer or two after work hours. Thanks also to all the guys on the volleyball teams in Trondheim and at MIT for three great years on the court and in the sand. Finally, a warm hug to my family for always being there with support and encouragement.

Jens Lohne Eftang
Trondheim, April 2011

CONTENTS

Introduction	1
Paper 1	
Evaluation of flux integral outputs for the reduced basis method	25
Paper 2	
An “ <i>hp</i> ” certified reduced basis method for parametrized elliptic partial differential equations	59
Paper 3	
An <i>hp</i> certified reduced basis method for parametrized parabolic partial differential equations	105
Paper 4	
A two-step certified reduced basis method	145
Paper 5	
Parameter multi-domain “ <i>hp</i> ” empirical interpolation	191
Paper 6	
<i>A posteriori</i> error bounds for the empirical interpolation method	219
Paper 7	
Approximation of parametric derivatives by the empirical interpolation method	231

INTRODUCTION

“Tallene og bokstavene møttes en dag og ville slåss. Vi vinner alltid, sa tallene. Vi gir oss aldri, sa bokstavene. Dermed lå de der hulter til bulter.”

Inger Hagerup

Partial differential equations (PDEs) describe the underlying physics in many problems from areas such as heat transfer, structural mechanics, fluid mechanics, or electromagnetics. Ever since the introduction of the modern computer there has been a growing interest in computer simulation of such physical systems. In parallel with the last decades' rapid increase in available computer power, there has been an impressive development of new and efficient numerical methods for computer simulation. Numerical methods for PDEs is one field of research that continues to be very active today.

One classical family of methods for the numerical approximation of PDEs is finite element (FE) methods. Depending on the particular problem at hand (for example regularity of the solution), a particular method (or class of methods) is typically better suited than others. A good method minimizes the computational time required to find a numerical approximation at a prescribed level of accuracy.

Most of the papers in this thesis are related to efficient numerical approximation of *parametrized* PDEs. In addition to the spatial and possibly temporal variables, a parametrized PDE depends on one or several parameters that enter into the equations as coefficients or through coefficient functions. These parameters may be related to physical properties of the system such as material properties or geometry, or interactions of the system with its environment such as applied forces or boundary conditions. The parameter dependence thus defines not only a single PDE but rather a family of PDEs which we wish to approximate. Parametrized PDEs are relevant in many engineering applica-

tions such as parameter estimation, design, or optimization; or for educational or scientific purposes to understand the behavior of the solution as a function of physical parameters.

Typically, one is interested in the solution to a parametrized PDE either for many different input parameter values — *many queries* — or the solution is required rapidly once the input parameter value is known — *real time*. In these contexts classical methods may be prohibitively expensive: they do not provide a sufficiently accurate solution within the time available for computation. However, the family of solutions induced by the underlying parameter dependence often provides an opportunity for *model reduction*: the solution associated with one parameter value is typically similar to the solution associated with a nearby parameter value. The reduced basis method [1, 33, 44] is one method that takes advantage of precisely this opportunity to provide computational speedup of solutions to parametrized PDEs without compromise to the accuracy of the approximation. The reduced basis method belongs to the large class of model order reduction methods. The common goals of such methods are to reduce the computational complexity of a given problem *and* preserve important properties of the system such as accuracy and stability.

This introduction chapter is organized as follows. First, we motivate and discuss in Section 1 model order reduction more generally. Then, in Section 2 we give an overview of the reduced basis method. Next, in Section 3 we give an overview of the related empirical interpolation method; this method is used for the approximation of parametrized functions, and is the focus of three of the papers in this thesis. Finally, in Section 4, we summarize and discuss the contributions from each of the papers.

1 Model Order Reduction

Classical numerical methods for PDEs, such as finite difference or finite element methods (FE) [9, 10, 35, 40], are frequently used to solve engineering problems in areas such as heat transfer, structural mechanics, electromagnetics, and fluid mechanics. Many of these problems can be solved efficiently with classical methods. However, for complicated problems the number of degrees of freedom required to resolve all features of the solution with sufficient accuracy may be very large. As a result, the associated computational cost may be very large and in some cases prohibitive.

When classical methods are too expensive, *model order reduction* methods may in some cases be used to reduce the number of degrees of freedom and as

Introduction

a result the required computational time. These methods attempt to replace the original (FE, say) system of high order — many degrees of freedom — with a reduced system of lower order — significantly fewer degrees of freedom — in such a way that the solution associated with the lower order system is a good approximation to the solution of the original higher order system. Model order reduction methods must thus in some sense identify the degrees of freedom that are important for the behavior of the system, and retain these degrees of freedom in the reduced system.

Many of the traditional model order reduction techniques have been developed within the field of dynamical system simulation and control. A full order system — a direct physical model or, say, the result of a highly accurate FE spatial discretization of a PDE — may be computationally too costly in the optimal control setting since the solution of the system is required for many (in advance unknown) input controls. For this reason methods for model order reduction have been considered. Classical approaches include Krylov subspace methods [3] and balanced truncation [30]. An approach related to balanced truncation which in fact for parabolic PDEs is relevant to the reduced basis context of this thesis [20], is proper orthogonal decomposition (POD) (also known as principal component analysis) [41, 43, 45]. All these model order reduction techniques share a common goal: provide significant reduction of the computational complexity and hence cost of the system *and* preserve important properties of the system such as accuracy and stability.

Another branch of model order reduction methods is found in the context of parametrized systems. In this case the input parameters may be physical properties of the system such as material properties, geometrical factors, or boundary conditions. Typically, the parameters enter as coefficients or through coefficient functions in a partial or ordinary differential equation that describe the physical system of interest.

In practical applications, the solution to the system — the state or the field variable — is typically not the main interest *per se*. Decisions in engineering are typically not based on how the solution “looks” but rather from more quantifiable measures that may be inferred from the state or the field. The main interest is thus certain *outputs of interest* derived from the solution, such as an average of the field variable over a small region.

Parametrized model order reduction is of particular interest in contexts such as optimal control, design, optimization, parameter estimation, or stochastic simulation. In these contexts either immediate output response is required, or the output is required for many different input parameters. Often, a full order system (many degrees of freedom) is prohibitively expensive in these contexts

— it may be impossible to solve the system to sufficient accuracy within the amount of time or on the computer hardware available. Model order reduction techniques on the other hand may reduce the number of degrees of freedom of the system such that sufficiently accurate output approximations are computationally feasible within these constraints.

2 The Reduced Basis Method

The first four papers in this thesis are directly related to the reduced basis (RB) method. In this section, we give a brief overview of the methodology, and a short introduction to some of the technical details involved.

2.1 Overview

The RB method is a computational and mathematical framework for parametric model order reduction of parametrized PDEs. The parameters enter into a parametrized PDE as coefficients or through coefficient functions that specify physical properties of the system or interactions between the system and the environment. The key observation of the RB method is that the solution to this equation resides on a typically low-dimensional manifold induced by the parameter dependence. When this manifold is smooth — the solution to the PDE varies smoothly with the parameters — it should be possible to reconstruct a good approximation of the solution associated with *any* parameter value with only limited knowledge of the manifold.

We now make this statement somewhat more precise. Let $\mathcal{D} \subset \mathbb{R}^P$ denote a predefined and bounded parameter domain, and let $\mu \in \mathcal{D}$ denote a particular parameter value; $P \geq 1$ is thus the number of parameters considered. For any $\mu \in \mathcal{D}$, we may write the parametrized PDE on the form

$$F_\mu(u(\mu)) = 0, \tag{2.1}$$

where F_μ is a parameter dependent (linear or non-linear) differential operator and $u(\mu)$ denotes the solution to the system. The parametric manifold on which the solution to this system resides is then given explicitly as

$$\mathcal{M} = \{u(\mu) : \mu \in \mathcal{D}\}. \tag{2.2}$$

Now, say that we are given N parameter values $\mu_1 \in \mathcal{D}, \dots, \mu_N \in \mathcal{D}$, and associated *snapshots* $u(\mu_1), \dots, u(\mu_N)$ of \mathcal{M} . If \mathcal{M} is smooth — $u(\mu)$ depends

Introduction

smoothly on the parameters — and μ_1, \dots, μ_N are well chosen, it is possible for *any* $\mu \in \mathcal{D}$ to compute a good approximation of $u(\mu)$ as a linear combination of the N snapshots $u(\mu_1), \dots, u(\mu_N)$. In particular, N may typically be chosen relatively small, and hence this approximation to $u(\mu)$ may be calculated at relatively low computational cost.

Of course, in general the snapshots $u(\mu_1), \dots, u(\mu_N)$ are not known exactly. In practice, we thus precompute highly accurate *truth* approximations, $u^{\mathcal{N}}(\mu_1) \approx u(\mu_1), \dots, u^{\mathcal{N}}(\mu_N) \approx u(\mu_N)$, where \mathcal{N} denotes the possibly very large number of degrees of freedom associated with these truth approximations. We shall consider here standard finite- or spectral element truth approximations [9, 10, 35, 40]. However, other variational frameworks such as discontinuous Galerkin methods [22] or finite volume methods [27] may be considered; see [13] and [20] for the application within the RB framework of discontinuous Galerkin methods and finite volume methods, respectively.

This “Lagrange” (snapshot-based) approach to the RB approximation seems to be the most popular approach in recent RB literature. However, alternative “Taylor” [33] or “Hermite” [24] approaches that include information about derivatives with respect to the parameters at one or more parameter values may also be considered. In this thesis, we exclusively pursue the “Lagrange” (snapshot-based) approach.

The RB method was originally developed in the context of non-linear structural analysis [1, 33]; the methodology was further considered for fluid flow problems in [24, 36]. Early work on error analysis of the RB approximation include [15, 37, 42]. More recent works have introduced

- rigorous *a posteriori* bounds for the error of the RB approximation with respect to the underlying truth FE discretization [38, 44];
- quasi-optimal sampling procedures for snapshot selection [20, 47]; and
- strict *offline-online* computational decoupling between a \mathcal{N} -dependent (expensive) precomputation stage followed by a \mathcal{N} -independent (much less expensive) input-output prediction stage [31, 44].

We shall discuss each of these ingredients briefly below and more extensively in the subsequent papers. The RB methodology is well developed for linear elliptic coercive and non-coercive equations [23, 44], linear parabolic equations [19, 20], and quadratically non-linear elliptic or parabolic equations such as the incompressible Navier-Stokes equations [25, 32, 46]. A comprehensive overview of both earlier and more recent contributions to the RB methodology can be found in [39, 44].

In particular with the recent development of rigorous *a posteriori* error bounds, quasi-optimal sampling algorithms, and strict offline-online computational decoupling procedures, the RB method may provide speedup of several orders of magnitude compared to the truth FE alternative with certifiable accuracy. The number of required RB snapshots (and thus degrees of freedom), N , is much smaller than the number of degrees of freedom associated with the truth FE space, \mathcal{N} . In addition, the error in the RB approximation can be rigorously bounded with respect to the truth FE discretization.

The rapid and certifiable RB output approximation is of particular interest in real time contexts such as parameter estimation [31] and optimal control [14, 19, 21] problems, and in many query contexts, such as stochastic [7] or multiscale [6, 26, 34] simulation. In both real time and many query contexts the significant precomputation effort required by the RB method can typically be justified. In the former the offline cost is often deemed irrelevant. In the latter, the offline cost is amortized over the many online evaluations; in particular, as the number of online evaluations $n \rightarrow \infty$, we obtain

$$\lim_{n \rightarrow \infty} \frac{\text{offline cost} + n(\text{online cost})}{n} = \text{online cost}. \quad (2.3)$$

2.2 Abstract Framework

The point of departure for the RB approximation is the weak form of the PDE. Let us now look at an example. Consider the particular parametrized differential operator F_μ over the spatial domain $\Omega \subset \mathbb{R}^d$, $d = 1, 2, 3$, given by

$$F_\mu(\cdot) = -\nabla \cdot (g(\mu)\nabla(\cdot)) - q, \quad (2.4)$$

where, for any $\mu \in \mathcal{D}$, $g(\mu) : \Omega \rightarrow \mathbb{R}$ is a parametrized coefficient function, and $q : \Omega \rightarrow \mathbb{R}$ is the load; we assume sufficient regularity of g and q . In this case the strong form of the PDE is

$$-\nabla \cdot (g(\mu)\nabla u(\mu)) = q, \quad \text{in } \Omega, \quad (2.5)$$

together with appropriate boundary conditions. We consider in this example

$$u(\mu) = 0, \quad \text{on } \Gamma_D \subset \partial\Omega, \quad (2.6)$$

$$\frac{\partial u(\mu)}{\partial n} = 0, \quad \text{on } \Gamma_N \subset \partial\Omega, \quad (2.7)$$

where $\partial\Omega = \Gamma_D \cup \Gamma_N$ is the boundary of Ω .

Introduction

We now introduce a function space V of test functions; in this case $V = H_D^1(\Omega)$, where $H_D^1(\Omega) = \{v \in H^1(\Omega) : v|_{\Gamma_D} = 0\}$, $H^1(\Omega) = \{v : |\nabla v| \in L^2(\Omega)\}$, and $L^2(\Omega)$ is the usual space of square integrable functions over Ω . We obtain the weak form of (2.5) by multiplication with a test function $v \in V$,

$$-\nabla \cdot (g(\mu)\nabla)u(\mu)v = qv, \quad \forall v \in V, \quad (2.8)$$

integration over Ω ,

$$-\int_{\Omega} \nabla \cdot (g(\mu)\nabla)u(\mu)v = \int_{\Omega} qv, \quad \forall v \in V, \quad (2.9)$$

and finally integration by parts,

$$\int_{\Omega} g(\mu)\nabla u(\mu) \cdot \nabla v = \int_{\Omega} qv + \int_{\partial\Omega} \frac{\partial u(\mu)}{\partial n} v, \quad \forall v \in V. \quad (2.10)$$

In this case the last term on the right hand side of (2.10) vanishes due to the boundary conditions (2.6) and (2.7) since either $v = 0$ or $\partial u(\mu)/\partial n = 0$ on $\partial\Omega$. The weak formulation of (2.5)–(2.7) then reads as follows: for any $\mu \in \mathcal{D}$, find $u(\mu) \in V$ such that

$$\int_{\Omega} g(\mu)\nabla u(\mu) \cdot \nabla v = \int_{\Omega} qv, \quad \forall v \in V. \quad (2.11)$$

More generally, we introduce a Hilbert space X associated with the exact solution of the problem. We further introduce for any $\mu \in \mathcal{D}$ a bilinear form $a(\cdot, \cdot; \mu) : X \times X \rightarrow \mathbb{R}$ and a linear functional $f(\cdot; \mu) : X \rightarrow \mathbb{R}$. The exact problem may then be stated as follows: For any $\mu \in \mathcal{D}$, find $u(\mu) \in X$ such that

$$a(u(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X. \quad (2.12)$$

We also introduce a linear and bounded *output functional* $\ell(\cdot; \mu) : X \rightarrow \mathbb{R}$. With the solution $u(\mu)$ of (2.12) we evaluate the output of interest through ℓ as

$$s(\mu) = \ell(u(\mu); \mu). \quad (2.13)$$

We note that if $a(\cdot, \cdot; \mu)$ is stable and continuous with respect to the X -norm, and $f(\cdot; \mu)$ is bounded with respect to the X -norm, $u(\mu) \in X$ exists and is unique [40].

For our particular example (2.11) we identify for all $w \in V$, $v \in V$,

$$a(w, v; \mu) = \int_{\Omega} g(\mu)\nabla w \cdot \nabla v, \quad f(v; \mu) = \int_{\Omega} qv. \quad (2.14)$$

If $g(\mu) > 0$ and $g(\mu)$ is bounded for all $\mu \in \mathcal{D}$, then $a(\cdot, \cdot; \mu)$ is coercive and continuous for all $\mu \in \mathcal{D}$. In this case, if q has sufficient regularity such that $f(\cdot; \mu)$ is bounded in V , existence of a unique solution $u(\mu)$ to the problem (2.11) for any $\mu \in \mathcal{D}$ is guaranteed by the Lax-Milgram Lemma [40].

2.3 Finite Element Discretization

We may now discretize (2.12) with (say) a classical FE method. To this end we introduce a discrete space $X^{\mathcal{N}} \subset X$ of finite dimension \mathcal{N} ; for example $X^{\mathcal{N}}$ may be the space of piecewise polynomials up to a specified polynomial order. The FE version of (2.12)–(2.13) then reads as follows: For any $\mu \in \mathcal{D}$, find $u^{\mathcal{N}}(\mu) \in X^{\mathcal{N}}$ such that

$$a(u^{\mathcal{N}}(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X^{\mathcal{N}}, \quad (2.15)$$

and evaluate the output

$$s^{\mathcal{N}}(\mu) = \ell(u^{\mathcal{N}}(\mu); \mu). \quad (2.16)$$

In the RB framework, we shall suppose that $X^{\mathcal{N}}$ is a high-fidelity FE space such that $u^{\mathcal{N}}$ is a good approximation to $u(\mu)$, and consequently $s^{\mathcal{N}}(\mu)$ is a good approximation to $s(\mu)$. Under this assumption we shall neglect the error $u(\mu) - u^{\mathcal{N}}(\mu)$ and consequently the output error $s(\mu) - s^{\mathcal{N}}(\mu)$: the FE solution $u^{\mathcal{N}}(\mu)$ and output $s^{\mathcal{N}}(\mu)$ are *truth approximations* to the corresponding exact quantities.

2.4 Reduced Basis Approximation

It is the computation of the truth approximation in (2.15)–(2.16) that we wish to accelerate by the RB method without loss of accuracy or rigor. To this end we introduce the RB approximation space

$$X_N = \text{span}\{u^{\mathcal{N}}(\mu_1), \dots, u^{\mathcal{N}}(\mu_N)\} \quad (2.17)$$

of dimension $N \ll \mathcal{N}$. The RB space is thus spanned by truth FE snapshots that reside on a presumably smooth parametric manifold $\mathcal{M}^{\mathcal{N}} = \{u^{\mathcal{N}}(\mu) : \mu \in \mathcal{D}\}$, as indicated in Figure 1.¹ We note that $\mathcal{M}^{\mathcal{N}}$ is only a small subset of the FE space $X^{\mathcal{N}}$; many of the \mathcal{N} degrees of freedom associated with the full FE space is

¹Note that in actual practice, an orthonormal basis for X_N is computed through a modified Gram-Schmidt procedure [16] to provide numerical stability.

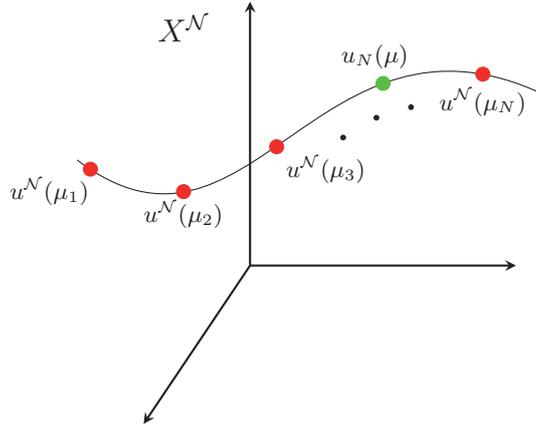


Figure 1: The RB approximation (green) is a linear combination of FE snapshots (red) that reside on a typically smooth parametric manifold. (This figure is a reproduction of Figure 16 in [44].)

thus in some sense redundant when attention is restricted only to this manifold. We expect that the RB space X_N accommodates good approximations $u_N(\mu) \approx u^{\mathcal{N}}(\mu)$ for any $\mu \in \mathcal{D}$ provided that the parameters μ_1, \dots, μ_N are well chosen.

The RB version of (2.15)–(2.16) now reads as follows: For any $\mu \in \mathcal{D}$, find $u_N(\mu) \in X_N$ such that

$$a(u_N(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_N \quad (2.18)$$

and evaluate the output

$$s_N(\mu) = \ell(u_N(\mu); \mu). \quad (2.19)$$

In addition, we may *certify* the RB solution or RB output prediction by rapid evaluation of RB *a posteriori* error estimators $\Delta_N(\mu)$ or $\Delta_N^{\text{out}}(\mu)$, such that

$$\Delta_N(\mu) \geq \|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_X, \quad (2.20)$$

or

$$\Delta_N^{\text{out}}(\mu) \geq |s^{\mathcal{N}}(\mu) - s_N(\mu)|, \quad (2.21)$$

respectively [44].

Clearly, the choice of the parameter values μ_1, \dots, μ_N is crucial to the quality of the RB approximation. In particular, if \mathcal{M}^N is smooth and μ_1, \dots, μ_N are judiciously chosen, we expect rapid convergence (as we increase N) of the RB approximation over \mathcal{D} .

Algorithm 1 Greedy sampling strategy

Set $N \leftarrow 1$ and choose $\mu_1 \in \mathcal{D}$ (randomly, say)
 Compute $u^N(\mu_1)$ and set $X_N = \text{span}\{u^N(\mu_1)\}$.
while $N < N_{\max}$ **do**
 $\mu_{N+1} = \arg \max_{\mu \in \Xi \subset \mathcal{D}} \Delta_N(\mu)$
 Compute $u^N(\mu_{N+1})$ and set $X_{N+1} = \text{span}\{u^N(\mu_1), \dots, u^N(\mu_{N+1})\}$.
 $N \leftarrow N + 1$
end while

In [47], the *Greedy sampling strategy* listed as Algorithm 1 was proposed. With this strategy, the RB space is constructed iteratively starting from only one (randomly chosen, say) parameter value $\mu_1 \in \mathcal{D}$ and associated (in practice normalized) basis function $u^N(\mu_1)$. At iteration N of the Greedy algorithm, the *a posteriori* error estimator $\Delta_N(\mu)$ is first evaluated over \mathcal{D} . (In practice, $\Delta_N(\mu)$ is evaluated over a finite *training set* $\Xi \subset \mathcal{D}$ — a rich computational surrogate for \mathcal{D} .) Denote by μ_{N+1} the particular parameter value at which the error estimator attains its maximum. The next RB space is then enriched with the (orthonormalized) snapshot $u^N(\mu_{N+1})$. We finally set $N \leftarrow N + 1$ and proceed to the next iteration; the procedure stops when a specified maximum RB space dimension N_{\max} is reached (or alternatively when the maximum of the *a posteriori* error estimator over Ξ reaches a specified tolerance).

In practice, the Greedy algorithm typically provides exponential convergence (with N) of the RB approximation over \mathcal{D} . It was first shown in [29] that for certain simple problems it is *possible* to choose the parameter values μ_1, \dots, μ_N such that exponential convergence is achieved. More recently, it is shown in [5, 8] that the Greedy algorithm achieves exponential convergence, if exponential convergence is possible.

We note that for time-dependent PDEs the snapshots must be chosen at judiciously chosen points in the combined time-parameter domain, and thus a modification to the standard Greedy sampling algorithm is necessary. A first extension of the Greedy algorithm to the time-dependent case was considered in [19]. More relevant to this thesis is a combined POD/Greedy procedure introduced in [20]; this procedure complements the standard Greedy procedure

Introduction

in the parameter domain with proper orthogonal decomposition in the temporal domain.

The offline-online computational decoupling is crucial to the efficiency of the RB method. The RB offline stage is essentially execution of the Greedy sampling algorithm: selection and computation of N finite element truth snapshots. In addition, an online dataset of size independent of \mathcal{N} is constructed in the offline stage and stored for subsequent evaluation in the online stage. The computational cost of this process is \mathcal{N} -dependent, and hence the offline stage may be rather expensive. However, the offline stage is performed only once as preprocessing.

The RB online stage is evaluation of the RB solution, RB output, and RB output error bound. Thanks to the online dataset constructed and stored in the offline stage, the cost of the RB online stage depends on N and *not* on \mathcal{N} [44].² Under the crucial assumption on parametric smoothness, N can be chosen significantly smaller than \mathcal{N} for acceptable numerical accuracy. The $N \times N$ algebraic system associated with the RB problem (2.18) is in this case significantly faster to solve than the $\mathcal{N} \times \mathcal{N}$ system associated with the FE problem (2.15). The RB approximation is thus useful in the many query or real time contexts.

3 The Empirical Interpolation Method

Papers 5, 6, and 7 of this thesis are related to the empirical interpolation method (EIM) [4, 18]. We now give an overview of the method and a brief review of the approximation procedure.

3.1 Overview

The EIM is a method for “affine” approximation of “non-affine” parameter dependent functions. The EIM was introduced in [4] as a tool within the RB framework for the approximation of non-affine differential operators. It was further considered for the approximation of non-linear differential operators in [18]; see also [17]. In the case of a linear differential operator, the efficient RB offline-online computational procedures require a differential operator that is

²In addition, this dataset enables efficient evaluation of the RB error bound in the *offline* stage. The training set $\Xi \subset \mathcal{D}$ in the Greedy sampling algorithm may thus be chosen rather large to allow exhaustive exploration of the parameter domain.

affine in functions of the parameter. In particular, for all $\mu \in \mathcal{D}$, the bilinear form $a(\cdot, \cdot; \mu)$ and the linear functional $f(\cdot; \mu)$ must admit expansions

$$a(\cdot, \cdot; \mu) = \sum_{q=1}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\mu), \quad (3.1)$$

$$f(\cdot; \mu) = \sum_{q=1}^{Q_f} f^q(\cdot) \Theta_f^q(\mu). \quad (3.2)$$

Here, for $1 \leq q \leq Q_a$, the $a^q : X \times X \rightarrow \mathbb{R}$ are parameter independent bilinear forms and the $\Theta_a^q : \mathcal{D} \rightarrow \mathbb{R}$ are parameter dependent functions, and for $1 \leq q \leq Q_f$, the $f^q : X \rightarrow \mathbb{R}$ are parameter independent linear functionals and the $\Theta_f^q : \mathcal{D} \rightarrow \mathbb{R}$ are parameter dependent functions. In particular, these affine expansions enable the offline construction of the online dataset (of size independent of \mathcal{N}), which in turn allows online evaluation of the RB solution, RB output, and RB output error bound at computational cost dependent only on N , Q_a , and Q_f , and *not* on \mathcal{N} . In the case of a non-affine differential operator, the EIM serves to construct an affine approximation and thus recover the efficient computational procedures.

In the case of a non-linear (in the solution $u(\mu)$) differential operator, it is only possible to develop efficient RB offline-online procedures for low-order polynomial non-linearities. In the case of higher order or non-polynomial non-linearities, the EIM serves to recover efficient offline-online procedures through approximations of the non-linear terms.

The EIM has applications also outside the RB framework; several examples are mentioned in [28]. A closely related method is the *discrete empirical interpolation method* [12], which combines a discrete version of the EIM with proper orthogonal decomposition for model order reduction of non-linear dynamical systems.

3.2 EIM Approximation

As in the previous section, Ω denotes here a spatial domain, and \mathcal{D} denotes a parameter domain. Consider the parametrized function $g : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$; we assume that $g(\cdot; \mu) \in L^\infty(\Omega)$ for all $\mu \in \mathcal{D}$. For any $\mu \in \mathcal{D}$, the EIM serves to construct an approximation $g_M(\cdot; \mu) \approx g(\cdot; \mu)$ such that $g_M(\cdot; \mu)$ resides in an approximation space

$$W_M = \text{span}\{g(\cdot; \mu_1), \dots, g(\cdot; \mu_M)\} \quad (3.3)$$

Introduction

of finite dimension M . We note that W_M is spanned by M snapshots of a parametric manifold

$$\mathcal{M}_g = \{g(\cdot; \mu) : \mu \in \mathcal{D}\}. \quad (3.4)$$

If \mathcal{M}_g is smooth, and if the parameters μ_1, \dots, μ_M are well chosen, it should thus be possible to find a good approximation $g_M(\cdot; \mu) \approx g(\cdot; \mu)$ in W_M for any $\mu \in \mathcal{D}$ for M relatively small. In practice, μ_1, \dots, μ_M are chosen through a Greedy sampling procedure.

For any $\mu \in \mathcal{D}$, the EIM approximation is given as a linear combination of the snapshots,

$$g_M(\cdot; \mu) = \sum_{m=1}^M \varphi_m(\mu) g(\cdot; \mu_m). \quad (3.5)$$

Here, the coefficients $\varphi_m(\mu)$, $1 \leq m \leq M$, are determined by *interpolation* at M judiciously chosen points $t_1 \in \Omega, \dots, t_M \in \Omega$. For $1 \leq m \leq M$, and all $\mu \in \mathcal{D}$, we thus determine $\varphi_m(\mu)$, $1 \leq m \leq M$, as the solution coefficients to the linear system

$$\sum_{m=1}^M \varphi_m(\mu) g(t_n; \mu_m) = g(t_n; \mu), \quad 1 \leq n \leq M. \quad (3.6)$$

Note that in contrast, the RB approximation is based on Galerkin *projection*.

We mention here the paper [28], which comprises a theoretical analysis of the EIM approximation and an elaborate discussion of the EIM interpolation nodes t_1, \dots, t_M . In particular, these nodes are compared with known optimal point distributions in the context of polynomial interpolation on simple domains. Because of a remarkable resemblance with optimal point distributions, the EIM interpolation nodes are in [28] called the *magic points*.

A typical use of the EIM is evaluation of parametrized integrals. Consider as an example

$$G(\mu) = \int_{\Omega} g(\cdot; \mu) \psi. \quad (3.7)$$

Here, $\psi : \Omega \rightarrow \mathbb{R}$, and we assume that the product $g(\cdot; \mu)\psi$ is integrable over Ω . (Note that a parametrized bilinear form associated with a parametrized partial differential equation may be viewed as a special case of (3.7).) Evaluation of $G(\mu)$ with for example standard Gauss-Lobatto-Legendre quadrature may be

prohibitively expensive, in particular when $G(\mu)$ is required in real time or for many different $\mu \in \mathcal{D}$. For any $\mu \in \mathcal{D}$ the EIM provides an approximation $G_M(\mu) \approx G(\mu)$ as

$$G_M(\mu) = \int_{\Omega} g_M(\cdot; \mu) \psi = \sum_{m=1}^M \varphi_m(\mu) \int_{\Omega} g(\cdot; \mu_m) \psi. \quad (3.8)$$

The key point here is that the integration can be performed as precomputation, since the integrals to the right are parameter independent.

4 Summary of Papers

4.1 Overview

The topics of this thesis are either directly or indirectly related to the reduced basis method. We now briefly describe the contributions from each of the papers.

A first topic is the evaluation of a particular class of output functionals in the reduced basis context: integrals of fluxes over parts of the boundary of the spatial domain. If we explicitly define the output functional ℓ as the integral of the flux, then ℓ is not an X -bounded functional. For this reason, flux integrals are typically not evaluated directly but rather indirectly through the variational problem formulation [2, 11]. This strategy necessitates the choice of a *lifting function*. In a standard FE context it is obvious how to choose this lifting function. In the RB context, this choice is not so obvious. In Paper 1 we comment on “good” choices for the lifting function and demonstrate the impact these choices have on the RB approximation error.

A second topic is the *hp reduced basis method* (*hp*-RB). This method is a generalization of the standard RB method, and may provide an additional online computational speedup of the RB approximation at additional offline cost. With this method we first construct an adaptive partition of the parameter domain \mathcal{D} into parameter subdomains; subsequently, we construct standard RB approximation spaces restricted to parameter values within each subdomain. The idea is that these local (in parameter) spaces may be chosen significantly smaller than the global spaces associated with a standard RB procedure.

With the *hp*-RB method we adopt the concepts of “*h*-type” and “*p*-type” approximations from the “FE language” to the parametric context of the RB method. The initial partition of the parameter domain is *h-refinement*; the subsequent application of the standard RB method within a parameter subdomain

Introduction

is *p-refinement*. The combination of these two strategies thus yields the *hp*-RB method. The *hp*-RB method is presented for elliptic equations in Paper 2 and for parabolic equations in Paper 3. In Paper 4 we introduce a *two-step RB method*. This method is of particular interest in the *hp*-RB context, since it may significantly reduce the *hp*-RB offline cost.

The remainder of the thesis is related to the empirical interpolation method. In Paper 5, we develop *hp* strategies for the empirical interpolation method. These *hp*-EIM methods share the fundamental ideas with the *hp*-RB method: partition of the full parameter domain into parameter subdomains, and application of the standard EIM procedure independently within each subdomain.

In the original EIM papers [4, 18], estimators for the EIM error were introduced. Although these estimators are typically very sharp, they are not rigorous error *bounds*. In Paper 6 we introduce rigorous *a posteriori* error bounds for the EIM approximation. These bounds work well and may be computed efficiently in certain simple cases. In the general case, more work is required in particular for the efficient computation of the bounds.

For a given parametrized function, the sharpness of the associated rigorous EIM error bounds introduced in Paper 6 depends on the error in the EIM approximation of derivatives of this parametrized function with respect to the parameters (parametric derivatives). This is the motivation for the work in Paper 7, in which we more generally consider the EIM approximation of parametric derivatives. In particular we show in this paper that, as we increase the number of EIM basis functions, the error in the EIM derivative approximation goes to zero if the error in the EIM approximation of the original function goes to zero.

Although the motivation for our work in Paper 7 was to assess the sharpness of the EIM error bounds in Paper 6, the results show that the EIM may be used in practice for the efficient approximation of parametric derivatives. Moreover, the convergence results in Paper 7 are in fact not restricted to the EIM, but apply to a much larger class of approximation schemes. The results may be useful in contexts such as optimization and parameter estimation, since Jacobians can be evaluated at only minor additional cost.

4.2 List of Papers

This thesis consists of this introduction chapter and the following papers:

Paper 1: J. L. Eftang and E. M. Rønquist, *Evaluation of flux integral outputs for the reduced basis method*. Mathematical Models and Methods in Applied

Sciences, Vol. 20, No. 3 (2010), pp. 351-374.

In this paper, we consider the evaluation of flux integral outputs from reduced basis solutions to second-order PDE's. In order to evaluate such outputs, a lifting function v^* must be chosen. In the standard finite element context, this choice is not relevant, whereas in the reduced basis context, as we show, it greatly affects the output error. We propose two "good" choices for v^* , and illustrate their effect on the output error by examining a numerical example. We also make clear the role of v^* in a more general primal-dual reduced basis approximation framework.

Paper 2: J. L. Eftang, A. T. Patera, and E. M. Rønquist. *An "hp" Certified Reduced Basis Method for Parametrized Elliptic Partial Differential Equations*. SIAM Journal on Scientific Computing, Vol. 32, No. 6 (2010), pp. 3170–3200.

We present a new "hp" parameter multidomain certified reduced basis (RB) method for rapid and reliable online evaluation of functional outputs associated with parametrized elliptic partial differential equations. We propose, and provide theoretical justification for, a new procedure for adaptive partition ("h"-refinement) of the parameter domain into smaller parameter subdomains: we pursue a hierarchical splitting of the parameter (sub)domains based on proximity to judiciously chosen parameter *anchor points* within each subdomain. Subsequently, we construct individual standard RB approximation spaces ("p"-refinement) over each subdomain. Greedy parameter sampling procedures and a posteriori error estimation play important roles in both the "h"-type and "p"-type stages of the new algorithm. We present illustrative numerical results for a convection-diffusion problem: the new "hp"-approach is considerably faster (respectively, more costly) than the standard "p"-type reduced basis method in the online (respectively, offline) stage.

Paper 3: J. L. Eftang, D. J. Knezevic, and A. T. Patera. *An hp Certified Reduced Basis Method for Parametrized Parabolic Partial Differential Equations*. To appear in Mathematical and Computer Modelling of Dynamical Systems, 2011.

In this paper we introduce an *hp* certified reduced basis method for parabolic partial differential equations. We invoke a POD (in time) / Greedy (in parameter) sampling procedure first in the initial partition of the parameter domain (*h*-refinement) and subsequently in the construction of reduced basis approximation spaces restricted to each parameter subdomain (*p*-refinement). We show that proper balance between additional POD modes and additional parameter values

Introduction

in the initial subdivision process guarantees convergence of the approach. We present numerical results for two model problems: linear convection-diffusion, and quadratically nonlinear Boussinesq natural convection. The new procedure is significantly faster (respectively, more costly) in the reduced basis Online (respectively, Offline) stage.

Paper 4: J. L. Eftang, D. B. P. Huynh, D. J. Knezevic, and A. T. Patera. *A Two-Step Certified Reduced Basis Method*. Accepted in Springer Journal of Scientific Computing, 2011.

In this paper we introduce a two-step Certified Reduced Basis (RB) method. In the first step we construct from an expensive finite element “truth” discretization of dimension \mathcal{N} an intermediate RB model of dimension $N \ll \mathcal{N}$. In the second step we construct from this intermediate RB model a *derived* RB (DRB) model of dimension $M \leq N$. The construction of the DRB model is effected at cost $\mathcal{O}(N)$ and in particular at cost independent of \mathcal{N} ; subsequent evaluation of the DRB model may then be effected at cost $\mathcal{O}(M)$. The DRB model comprises both the DRB output *and* a rigorous *a posteriori* error bound for the error in the DRB output with respect to the *truth* discretization.

The new approach is of particular interest in two contexts: *focus calculations* and *hp-RB approximations*. In the former the new approach serves to reduce online cost, $M \ll N$: the DRB model is restricted to a slice or subregion of a larger parameter domain associated with the intermediate RB model. In the latter the new approach enlarges the class of problems amenable to *hp*-RB treatment by a significant reduction in offline (precomputation) cost: in the development of the *hp* parameter domain partition and associated “local” (now derived) RB models the finite element truth is replaced by the intermediate RB model. We present numerical results to illustrate the new approach.

Paper 5: J. L. Eftang and B. Stamm. *Parameter Multi-Domain “hp” Empirical Interpolation*. NTNU Preprint Numerics No. 3/2011. Submitted to International Journal for Numerical Methods in Engineering, 2011.

In this paper, we introduce two parameter multi-domain “*hp*” techniques for the empirical interpolation method (EIM). In both approaches, we construct a partition of the original parameter domain into parameter subdomains: *h*-refinement. We apply the standard EIM independently within each subdomain to yield local (in parameter) approximation spaces: *p*-refinement. Further, for a particularly simple case we introduce *a priori* convergence theory for the partition procedure. We show through two numerical examples that our approaches

provide significant reduction in the EIM approximation space dimension, and thus significantly reduce the computational cost associated with EIM approximations.

Paper 6: J. L. Eftang, M. A. Grepl, and A. T. Patera. *A Posteriori Error Bounds for the Empirical Interpolation Method*. *Comptes Rendus Mathématique*, Vol. 348, No. 9-10 (2010), pp. 575-579.

We present rigorous *a posteriori* error bounds for the Empirical Interpolation Method (EIM). The essential ingredients are (i) analytical upper bounds for the parametric derivatives of the function to be approximated, (ii) the EIM “Lebesgue constant,” and (iii) information concerning the EIM approximation error at a finite set of points in parameter space. The bound is computed “offline” and is valid over the entire parameter domain; it is thus readily employed in (say) the “online” reduced basis context. We present numerical results that confirm the validity of our approach.

Paper 7: J. L. Eftang, M. A. Grepl, E. M. Rønquist, and A. T. Patera. *Approximation of Parametric Derivatives by the Empirical Interpolation Method*. NTNU Preprint Numerics No. 4/2011. To be submitted to *Foundations of Computational Mathematics*, 2011.

We introduce a general *a priori* convergence result for the approximation of parametric derivatives. We show — with rather general assumptions on the particular approximation scheme — that the derivative approximation is convergent provided that the approximation to the original function is convergent. In this paper we focus on the approximation of parametric derivatives by the Empirical Interpolation Method (EIM); we present numerical results with the EIM to illustrate the general theory.

Bibliography

- [1] B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16:525–528, 1978.
- [2] I. Babuška and A. Miller. The post-processing approach in the finite element method—part 1: Calculation of displacements, stresses and other higher derivatives of the displacements. *International Journal for Numerical Methods in Engineering*, 20(6):1085–1109, 1984.

Introduction

- [3] Z. Bai. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Applied Numerical Mathematics*, 43(1-2):9 – 44, 2002.
- [4] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [5] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. Technical Report 5, Interdisciplinary Mathematics Institute, College of Arts and Sciences, University of South Carolina, 2010.
- [6] S. Boyaval. Reduced-basis approach for homogenization beyond the periodic setting. *Multiscale Modeling & Simulation*, 7(1):466–494, 2008.
- [7] S. Boyaval, C. Le Bris, Y. Maday, N. C. Nguyen, and A. T. Patera. A reduced basis approach for variational problems with stochastic parameters: Application to heat conduction with variable robin coefficient. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44):3187 – 3206, 2009.
- [8] 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. *ESAIM: Mathematical Modelling and Numerical Analysis*, submitted, November 2009. <http://augustine.mit.edu>.
- [9] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral methods. Fundamentals in single domains*. Scientific Computation. Springer-Verlag, Berlin, 2006.
- [10] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral methods. Evolution to complex geometries and applications to fluid dynamics*. Scientific Computation. Springer, Berlin, 2007.
- [11] G. F. Carey. Derivative calculation from finite element solutions. *Computer Methods in Applied Mechanics and Engineering*, 35(1):1–14, 1982.
- [12] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

- [13] Y. Chen, J. S. Hesthaven, Y. Maday, and J. Rodríguez. Certified reduced basis methods and output bounds for the harmonic maxwell's equations. *SIAM Journal on Scientific Computing*, 32(2):970–996, 2010.
- [14] L. Dedè. Reduced basis method and error estimation for parametrized optimal control problems with control constraints. *Journal of Scientific Computing*, 2011. 10.1007/s10915-011-9483-5.
- [15] J. P. Fink and W. C. Rheinboldt. On the error behavior of the reduced basis technique for nonlinear finite element approximations. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 63(1):21–28, 1983.
- [16] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.
- [17] M. Grepl. *A Posteriori Error Bounds for Reduced-Basis Approximations of Nonaffine and Nonlinear Parabolic Partial Differential Equations*. *Mathematical Models and Methods in Applied Sciences*, submitted, 2010.
- [18] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [19] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005.
- [20] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
- [21] B. Haasdonk and M. Ohlberger. Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Mathematical and Computer Modelling of Dynamical Systems*, 2011. 10.1080/13873954.2010.514703.
- [22] J. S. Hesthaven and T. Warburton. *Nodal discontinuous Galerkin methods*, volume 54 of *Texts in Applied Mathematics*. Springer, New York, 2008. Algorithms, analysis, and applications.

Introduction

- [23] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique*, 345(8):473 – 478, 2007.
- [24] K. Ito and S. S. Ravindran. A reduced-order method for simulation and control of fluid flows. *Journal of Computational Physics*, 143(2):403 – 425, 1998.
- [25] D. J. Knezevic, N. C. Nguyen, and AT Patera. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, *accepted*, 2010, August.
- [26] D. J. Knezevic and A. T. Patera. A certified reduced basis method for the fokker–planck equation of dilute polymeric fluids: Fene dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793–817, 2010.
- [27] D. Kröner. *Numerical schemes for conservation laws*. Wiley-Teubner Series Advances in Numerical Mathematics. John Wiley & Sons Ltd., Chichester, 1997.
- [28] Y. Maday, N. C. Nguyen, A. T. Patera, and G. S. H. Pau. A general multipurpose interpolation procedure: The magic points. *Communications in Pure and Applied Mathematics*, 8:383–404, 2009.
- [29] Y. Maday, A. T. Patera, and G. Turinici. Global a priori convergence theory for reduced-basis approximations of single-parameter symmetric coercive elliptic partial differential equations. *Comptes Rendus Mathematique*, 335(3):289 – 294, 2002.
- [30] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17 – 32, February 1981.
- [31] N. C. Nguyen, G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes: Application to real-time bayesian parameter estimation. In Biegler, Birosoa, Ghattas, Heinkenschloss, Keyes, Mallick, Marzouk, Tenorio, Waanders, and Willcox, editors, *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 151–177. John Wiley & Sons, Ltd, 2010.

- [32] N. C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous burgers' equation. *Calcolo*, 46:157–185, 2009.
- [33] A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18:455–462, 1980.
- [34] M. Ohlberger and K. Smetana. A new problem adapted hierarchical model reduction technique based on reduced basis methods and dimensional splitting. Preprint 03/10, FB 10, University Muenster, december 2010.
- [35] A. T. Patera. A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *Journal of Computational Physics*, 54(3):468 – 488, 1984.
- [36] J. S. Peterson. The reduced basis method for incompressible viscous flow calculations. *SIAM Journal on Scientific and Statistical Computing*, 10(4):777–786, 1989.
- [37] T. A. Porsching. Estimation of the error in the reduced basis method solution of nonlinear equations. *Mathematics of Computation*, 45(172):487–496, 1985.
- [38] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici. Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *Journal of Fluids Engineering*, 124(1):70–80, 2002.
- [39] A. Quarteroni. *Numerical models for differential problems*, volume 2 of *MS&A. Modeling, Simulation and Applications*. Springer-Verlag Italia, Milan, 2009. Translated from the 4th (2008) Italian edition by Silvia Quarteroni.
- [40] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*, volume 23 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1994.
- [41] S. S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34(5):425–448, 2000.

Introduction

- [42] W. C. Rheinboldt. On the theory and error estimation of the reduced basis method for multi-parameter problems. *Nonlinear Analysis*, 21(11):849–858, 1993.
- [43] C. W. Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, 15(3):997–1013, 2005.
- [44] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [45] L. Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.
- [46] K. Veroy and A. T. Patera. Certified real-time solution of the parametrized steady incompressible navier-stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005.
- [47] K. Veroy, C. Prud’homme, D. V. Rovas, and A. T. Patera. 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.

PAPER 1

**EVALUATION OF FLUX INTEGRAL OUTPUTS FOR
THE REDUCED BASIS METHOD**

JENS L. EFTANG AND EINAR M. RØNQUIST

Published in
Mathematical Models and Methods in Applied Sciences,
Vol. 20, No. 3 (2010), pp. 351-374

EVALUATION OF FLUX INTEGRAL OUTPUTS FOR THE REDUCED BASIS METHOD

JENS L. EFTANG AND EINAR M. RØNQUIST

*Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

Abstract

In this paper, we consider the evaluation of flux integral outputs from reduced basis solutions to second-order PDE's. In order to evaluate such outputs, a lifting function v^* must be chosen. In the standard finite element context, this choice is not relevant, whereas in the reduced basis context, as we show, it greatly affects the output error. We propose two “good” choices for v^* , and illustrate their effect on the output error by examining a numerical example. We also make clear the role of v^* in a more general primal-dual reduced basis approximation framework.

1 Introduction

For many practical applications, one is interested in certain physical averages, or *outputs of interest*, which can be defined as functionals of the solution to a partial differential equation (PDE) that describes an underlying physical problem. For example, the output of interest may be the average heat flux through (or average temperature on) a surface, or the average force acting on a wall due to fluid flow. In this paper, we are concerned with outputs of *flux integral type*, i.e., outputs that can be written as surface integrals of the normal derivative of the solution to the underlying PDE. We consider second-order equations, for which it is possible to evaluate flux integral outputs directly via the weak problem formulation, and in particular without the need for any numerical differentiation.

Mathematically, we consider a weakly written problem defined on a domain Ω : Find $u \in X(\Omega)$ such that

$$a(u, v) = f(v), \quad \forall v \in X(\Omega), \quad (1.1)$$

where a is a coercive, continuous and for simplicity also symmetric bilinear form derived from some second-order differential operator, f is a linear and bounded

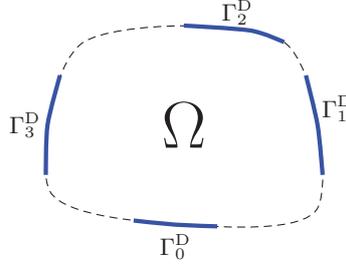


Figure 1: Illustration of a general domain Ω in the particular case that the Dirichlet boundary $\Gamma^{\text{D}} = \cup_{k=0}^{K-1} \Gamma_k^{\text{D}}$ consists of $K = 4$ disjoint sections.

functional, $X(\Omega) = \{v \in H^1(\Omega) : v|_{\Gamma^{\text{D}}} = 0\}$ is our exact space, and $\Gamma^{\text{D}} \subset \partial\Omega$ denotes the parts of the boundary of Ω on which we impose (for simplicity homogeneous) Dirichlet boundary conditions. As usual, $H^1(\Omega)$ denotes the Sobolev space of functions with square integrable first order derivatives over Ω . Henceforth, the Ω -dependence of our spaces is understood when no ambiguity may arise.

We shall assume that the Dirichlet boundary, Γ^{D} , may be written as $\Gamma^{\text{D}} = \cup_{k=0}^{K-1} \Gamma_k^{\text{D}}$, where $\Gamma_k^{\text{D}} \subset \partial\Omega$, $0 \leq k \leq K-1$, are disjoint sections, as illustrated in Figure 1 for the special case $K = 4$. In addition, we require any two such sections Γ_i^{D} and Γ_j^{D} ($i \neq j$) to be separated by a section on which a Neumann boundary condition is imposed. Our output of interest shall be the integral of the flux through $\Gamma_0^{\text{D}} \subseteq \Gamma^{\text{D}}$, i.e., the integral of the flux through an entire separate section of Γ^{D} . We thus define the output functional

$$\tilde{l}^{\text{out}}(w) \equiv \int_{\Gamma_0^{\text{D}}} \frac{\partial w}{\partial n} \, ds, \quad (1.2)$$

where $\partial/\partial n$ denotes the outward normal derivative and s is the surface measure on $\partial\Omega$. When solving e.g. Poisson or Helmholtz problems with the finite element (FE) method, it is preferable [1, 2, 6, 10] to instead evaluate flux integral outputs through an affine functional

$$l^{\text{out}}(w) \equiv a(w, v^*) - f(v^*), \quad (1.3)$$

where $v^* \in H^1$ is any function that is equal to unity on Γ_0^{D} and equal to zero on $\Gamma^{\text{D}} \setminus \Gamma_0^{\text{D}}$. Of course, even though (1.2) and (1.3) make sense for any $w \in X$, they are only of interest for $w \approx u$, where u is the solution of (1.1). One way

Evaluation of Flux Integral Outputs

to derive (1.3), is to recast the original problem (1.1) as a “Neumann problem” for which there are no restrictions on the test and trial functions on Γ_0^D . Thus, if we suppose (1.1) is a Poisson or Helmholtz problem, this modified problem reads: Find $u \in \tilde{X}$ such that

$$a(u, v) = f(v) + \int_{\Gamma_0^D} \frac{\partial u}{\partial n} v \, ds, \quad \forall v \in \tilde{X}, \quad (1.4)$$

where $\tilde{X} = \{v \in H^1 : v|_{\Gamma^D \setminus \Gamma_0^D} = 0\} \supset X$. Moving $f(v)$ to the left hand side and choosing $v = v^* \in \tilde{X}$, we see that $\tilde{l}^{\text{out}}(u) = l^{\text{out}}(u)$.

Suppose we solve (1.1) numerically to obtain a FE approximation to u , $u^{\mathcal{N}} \in X^{\mathcal{N}}$, satisfying

$$a(u^{\mathcal{N}}, v) = f(v), \quad \forall v \in X^{\mathcal{N}}. \quad (1.5)$$

Here, $X^{\mathcal{N}} \subset X$ is a discrete FE space of dimension \mathcal{N} . The FE output of interest can now be evaluated in two ways, either as $\tilde{l}^{\text{out}}(u^{\mathcal{N}})$ or as $l^{\text{out}}(u^{\mathcal{N}})$. In the latter case, we substitute $u^{\mathcal{N}}$ for u and consequently \approx for $=$ in (1.4). With $v = v^*$, we get $l^{\text{out}}(u^{\mathcal{N}}) \approx \int_{\Gamma_0^D} \frac{\partial u^{\mathcal{N}}}{\partial n} \, ds$. Hence, in general, $l^{\text{out}}(u^{\mathcal{N}}) \neq \tilde{l}^{\text{out}}(u^{\mathcal{N}})$.

We shall refer to v^* as a *flux lifting function*, and we shall denote the set of possible such functions as V^* . Specifically,

$$V^* \equiv \{v \in H^1 : v|_{\Gamma^D \setminus \Gamma_0^D} = 0, v|_{\Gamma_0^D} = 1\}. \quad (1.6)$$

In [2], v^* is called an *extraction function*, and the method described above for flux integral output evaluation is an example of an *extraction method*. In [10], the method — with more emphasis on pointwise quantities, rather than on surface integrals — is called *the consistent Galerkin FEM*. A collection of *post-processing methods* for flux integral and other types of outputs, including pointwise quantities, can be found in [1].

Typically, $l^{\text{out}}(u^{\mathcal{N}})$ converges to $l^{\text{out}}(u)$ quadratically with the energy-norm error of the field variable. In contrast, $\tilde{l}^{\text{out}}(u^{\mathcal{N}})$ typically exhibits only linear convergence [2, 6]. Another advantage of l^{out} over \tilde{l}^{out} is that the former requires no calculation of normal derivatives, which is particularly convenient in higher dimensions and for problems on domains with curved boundaries.

Aside from the essential boundary condition in (1.6), we have not made any particular choice for $v^* \in H^1$. In fact, within a standard finite element framework, this choice is not a big issue due to Galerkin orthogonality and the richness of the approximation spaces used [2, 6]. In contrast, as we will show

numerically and theoretically, one should take a little more care when evaluating flux integral outputs by way of the method described above within the *reduced basis* (RB) framework. For a thorough introduction to RB methods, confer e.g. [17] or [20].

In the next section, we shall consider a very simple numerical example which illustrates how l^{out} may be superior to \tilde{l}^{out} in terms of numerical accuracy within the FE framework. In Section 3, we first briefly review the RB method and then elaborate on the discrepancy between the FE and RB methods with respect to the choice of v^* . We then propose two “good” choices for v^* to use in the RB context. We also make clear the role of v^* in the more general primal-dual RB approximation procedure which is used to speed up the convergence for non-compliant problems [18, 20]. In Section 4, we remain in the RB context and illustrate the effect of different v^* ’s by examining yet another numerical example, and in Section 5 we end our discussion with some concluding remarks.

2 Flux Output Evaluation: a 1D Example

We consider a one-dimensional Helmholtz problem on $\Omega = (0, 2)$ with homogeneous Dirichlet boundary conditions. The weak formulation of the problem reads: Find $u \in H_0^1$ such that

$$\underbrace{\int_0^2 \left(\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + uv \right) dx}_{=a(u,v)} = \underbrace{\int_0^2 qv dx}_{=f(v)}, \quad \forall v \in H_0^1, \quad (2.1)$$

where $H_0^1 = \{v \in H^1 : v(0) = v(2) = 0\}$. For the purpose of this example, we want the solution u to be weakly singular. In order to achieve this, we choose the source term as $q(x) = x^{2/3}$. Our output of interest is the derivative of u at $x = \Gamma_0^{\text{D}} = \{2\}$, and our two output functionals now reduce to

$$\tilde{l}^{\text{out}}(w) = \left. \frac{\partial w}{\partial x} \right|_{x=2} \quad (2.2)$$

and

$$l^{\text{out}}(w) = a(w, v^*) - f(v^*), \quad (2.3)$$

where $v^* \in V^*$.

Evaluation of Flux Integral Outputs

With a spectral (high order polynomial) method, we discretise (2.1) and find $u^{\mathcal{N}} \in X^{\mathcal{N}}$ such that

$$a(u^{\mathcal{N}}, v) = f(v), \quad \forall v \in X^{\mathcal{N}}, \quad (2.4)$$

where

$$X^{\mathcal{N}} = \{v \in \mathbb{P}_{\mathcal{N}} : v_{\Gamma^{\text{D}}} = 0\} \quad (2.5)$$

is our discrete space. Here, $\Gamma^{\text{D}} = \{0, 2\}$ and $\mathbb{P}_{\mathcal{N}}$ denotes the space of polynomials of degree \mathcal{N} (note that here, $\dim(X^{\mathcal{N}}) = \mathcal{N} - 1$ due to the Dirichlet boundary conditions).

We shall also consider the dual problem: Find $\psi \in H_0^1$ such that

$$a(v, \psi) = -a(v, v^*), \quad \forall v \in H_0^1. \quad (2.6)$$

The spectral discretisation of (2.6) reads: Find $\psi^{\mathcal{N}} \in X^{\mathcal{N}}$ such that

$$a(v, \psi^{\mathcal{N}}) = -a(v, v^*), \quad \forall v \in X^{\mathcal{N}}. \quad (2.7)$$

Note that $a(v, v^*)$ — the (bounded) linear functional part of l^{out} — enters on the right hand side in the dual problem (with a minus sign). Thus, v^* also plays the role of a Dirichlet lifting function for the dual problem, with Dirichlet data equal to unity on Γ_0^{D} (i.e. at $x = 2$). Also note that the dual problem exhibits no (singular) source term. Provided v^* is smooth (deliberately choosing v^* singular seems somewhat peculiar), we expect ψ to be a smooth function and thus the convergence of $\psi^{\mathcal{N}}$ to ψ to be of infinite order.

We are interested in the errors in the output of interest, which we define for each of our two output functionals as

$$\tilde{e}^{\mathcal{N}, \text{out}} \equiv \left| \frac{\partial u}{\partial x} \Big|_{x=2} - \tilde{l}^{\text{out}}(u^{\mathcal{N}}) \right|, \quad (2.8)$$

and

$$e^{\mathcal{N}, \text{out}} \equiv \left| \frac{\partial u}{\partial x} \Big|_{x=2} - l^{\text{out}}(u^{\mathcal{N}}) \right| = \left| l^{\text{out}}(u) - l^{\text{out}}(u^{\mathcal{N}}) \right|, \quad (2.9)$$

respectively. For $e^{\mathcal{N}, \text{out}}$, we deduce that

$$\begin{aligned} e^{\mathcal{N}, \text{out}} &= |a(u - u^{\mathcal{N}}, v^*)| \\ &= |a(u - u^{\mathcal{N}}, \psi)| \\ &= |a(u - u^{\mathcal{N}}, \psi - \psi^{\mathcal{N}})| \\ &\leq \|u - u^{\mathcal{N}}\| \|\psi - \psi^{\mathcal{N}}\|, \end{aligned} \quad (2.10)$$

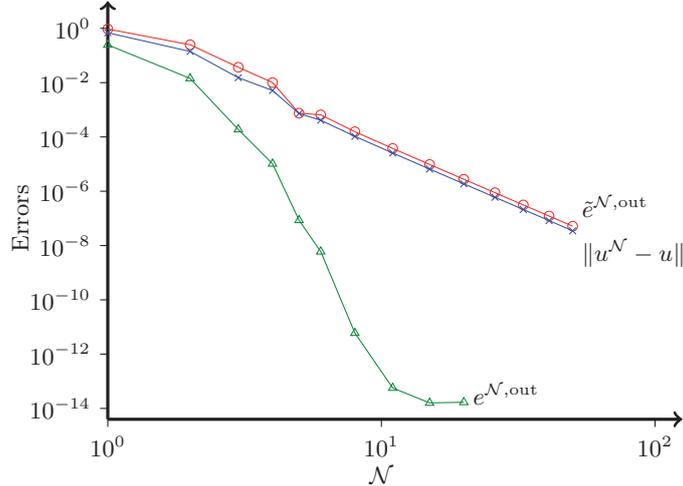


Figure 2: Energy error (\times) and output errors $e^{\mathcal{N}, \text{out}}$ (\triangle) and $\tilde{e}^{\mathcal{N}, \text{out}}$ (\circ) for increasing polynomial degree, \mathcal{N} , of the underlying spectral approximation.

by using the definition of $e^{\mathcal{N}, \text{out}}$, the fact that $u - u^{\mathcal{N}} \in H_0^1$ and the definition of the dual problem (2.6), Galerkin orthogonality of $u - u^{\mathcal{N}}$ and the Cauchy-Schwarz inequality, respectively. Here, $\|\cdot\| = \sqrt{a(\cdot, \cdot)}$ denotes the energy norm. A consequence of this estimate is that if ψ happens to be a smooth function, $e^{\mathcal{N}, \text{out}}$ will decay exponentially with \mathcal{N} , even if u is singular. Note that in practice, we never actually compute the discrete solution $\psi^{\mathcal{N}}$ to the dual problem.

In Figure 2, we plot the energy norm error $\|u - u^{\mathcal{N}}\|$ and the output errors $\tilde{e}^{\mathcal{N}, \text{out}}$ and $e^{\mathcal{N}, \text{out}}$ for $1 \leq \mathcal{N} \leq 50$. As our flux lifting function, we have made the choice $v^* = x/2$. As expected, the error $\|u - u^{\mathcal{N}}\|$ decays algebraically, while $e^{\mathcal{N}, \text{out}}$ decays at an infinite rate due to the smoothness of ψ . Note that we reach double precision accuracy for $e^{\mathcal{N}, \text{out}}$ for $\mathcal{N} \gtrsim 10$.

Let us make a few remarks concerning the above results. Firstly, although for $v, w \in X^{\mathcal{N}}$ the integrals $a(v, w)$ and $f(v)$ are easy to evaluate analytically, we have chosen to perform all integration by using numerical quadrature since we can then employ a very general computational framework (which can also be used to solve more difficult problems in which analytic integration is not possible). However, since $q(x) = x^{2/3}$ is not a smooth function over Ω , the integrand on the right hand side of the primal problem is singular, and hence accurate evaluation of the integral $f(v)$ requires quadrature of very high order.

Evaluation of Flux Integral Outputs

If $f(v)$ is not computed with sufficient accuracy, $u^{\mathcal{N}}$ will carry an additional numerical integration error which will compromise Galerkin orthogonality — which we exploited in the error bound (2.10) — and hence also the exponential convergence of the output. In our numerical experiment, we assume negligible numerical integration errors since all numerical integration is performed with Gauss-Lobatto-Legendre (GLL) quadrature [5] over $n_q \gg \mathcal{N} + 1$ quadrature points. Specifically, $n_q = 121$.

Secondly, we note that the error $\|u - u^{\mathcal{N}}\|$ decays much faster than expected from the standard error estimate. It can be shown that, for our problem, the most singular term of u behaves like $x^{8/3}$, and hence $u \in H^\sigma$ for all $0 \leq \sigma < 19/6$. From the standard estimate [5], we would expect $\|u - u^{\mathcal{N}}\| \leq c\mathcal{N}^{1-\sigma}$ for some positive constant c , and hence a slope of $-13/6$. In contrast, the actual decay is in fact of order -4.33 , which is twice as good as expected. The reason for this superconvergence is related to the particular type of singularity exhibited by the solution to our problem — it appears at an endpoint and is of x^α -type — and the rather good capability of polynomials to approximate functions with such singularities [4, 12].

Thirdly, we note that for our simple one-dimensional problem, $\tilde{e}^{\mathcal{N},\text{out}}$ decays as fast as the error $\|u - u^{\mathcal{N}}\|$. We would expect similar results were we to use a linear (h -type) finite element method. In fact, if u_h denotes a linear FE approximation to u on a mesh with elements of size h , we would from standard FE error estimates [19] expect convergence of order $\mathcal{O}(h^2)$ for the output $l^{\text{out}}(u_h)$, and of order $\mathcal{O}(h)$ for the energy-norm error $\|u - u_h\|$ (note that for our particular problem, the singularity in u is weak enough that full linear convergence (in h) is achieved). Based on the preceding results, we would also expect convergence of order $\mathcal{O}(h)$ for the output $\tilde{l}^{\text{out}}(u_h)$. Indeed, our results and comments here are consistent with results presented in [6] and a note made in [2] for the h and h - p finite element methods, respectively, applied to the problem (2.4).

Finally, as noted in [2], any two choices of $v^* \in (V^* \cap \tilde{X}^{\mathcal{N}})$ produce the same result for $l^{\text{out}}(u^{\mathcal{N}})$. To see this, let $v_1^*, v_2^* \in (V^* \cap \tilde{X}^{\mathcal{N}})$. Then $w^* = v_1^* - v_2^* \in X^{\mathcal{N}}$, and

$$[a(u^{\mathcal{N}}, v_1^*) - l(v_1^*)] - [a(u^{\mathcal{N}}, v_2^*) - l(v_2^*)] = a(u^{\mathcal{N}}, w^*) - l(w^*) = 0, \quad (2.11)$$

by (2.4). A convenient choice for v^* , then, is the function that is equal to unity at the node at $x = 2$ and equal to zero at every other node (or, in the low-order finite element case, the function that is equal to unity at $x = 2$ with support only on the element adjacent to the boundary). For this reason, we do

not specifically emphasise our choice for v^* when considering outputs from FE solutions.

In the next section, we turn our focus to flux integral outputs in the reduced basis context.

3 Flux Output Evaluation in the RB Framework

3.1 RB formulation

3.1.1 Parametrised weak form

Of interest within the reduced basis (RB) framework [9, 18, 20] is the solution — and ultimately the corresponding output of interest — of parameter-dependent PDE's in cases where the output is either required in real time, or for a large number of input parameters. In the following, $\mathcal{D} \subset \mathbb{R}^P$ shall denote the admissible parameter domain, and $\boldsymbol{\mu} \in \mathcal{D}$ a parameter P -tuple that governs e.g. boundary conditions or material or geometrical properties of the underlying physical problem. We shall consider the following parametrised problem on a domain Ω : Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u(\boldsymbol{\mu}) \in X$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}), \quad \forall v \in X, \quad (3.1)$$

where, for each $\boldsymbol{\mu} \in \mathcal{D}$, $a(\cdot, \cdot; \boldsymbol{\mu})$ is a coercive, continuous and for simplicity also symmetric bilinear form originating from a second-order differential operator, and $f(\cdot; \boldsymbol{\mu})$ is a linear and bounded functional. We also assume that a and f are parametrically affine, in the sense that the $\boldsymbol{\mu}$ -dependency of a and f take the form

$$a(v, w; \boldsymbol{\mu}) = \sum_{q=1}^{Q_a} a^q(v, w) \Theta_a^q(\boldsymbol{\mu}), \quad f(v; \boldsymbol{\mu}) = \sum_{q=1}^{Q_f} f^q(v) \Theta_f^q(\boldsymbol{\mu}), \quad (3.2)$$

for finite numbers Q_a and Q_f , where the a^q and f^q are parameter independent bilinear and linear forms, respectively, and the Θ_a^q and Θ_f^q are parameter dependent functions.

We still impose homogeneous Dirichlet boundary conditions on $\Gamma^D \subset \partial\Omega$, and take as our exact output of interest the flux integral

$$\tilde{s}(\boldsymbol{\mu}) = \int_{\Gamma_0^D} \frac{\partial u(\boldsymbol{\mu})}{\partial n} \, ds, \quad (3.3)$$

Evaluation of Flux Integral Outputs

where $\Gamma_0^D \subseteq \Gamma^D$ is an entire separate section of the Dirichlet boundary.

Finally, we define the now parameter dependent “energy” norm

$$\|\cdot\|_{\boldsymbol{\mu}} \equiv \sqrt{a(\cdot, \cdot; \boldsymbol{\mu})}, \quad (3.4)$$

and the equivalent parameter-independent X -norm

$$\|\cdot\|_X \equiv \sqrt{a(\cdot, \cdot; \bar{\boldsymbol{\mu}})}, \quad (3.5)$$

where $\bar{\boldsymbol{\mu}} \in \mathcal{D}$ is some fixed, preselected reference parameter. Note that by the assumptions of symmetry, coercivity and continuity, $a(\cdot, \cdot; \boldsymbol{\mu})$ defines an inner-product, and $\|\cdot\|_{\boldsymbol{\mu}}$ a norm equivalent to the H_1 -norm, for each $\boldsymbol{\mu} \in \mathcal{D}$.

3.1.2 FE “truth” approximation

The reduced basis approximations will be built upon snapshots of “truth” finite element approximations to solutions of (3.1), computed for wisely selected points in the admissible parameter domain. Let $X^{\mathcal{N}}$ be a standard finite element discrete space of dimension \mathcal{N} . The “truth” discretisation of (3.1) reads: For any $\boldsymbol{\mu} \in \mathcal{D}$, find $u^{\mathcal{N}}(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ such that

$$a(u^{\mathcal{N}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}), \quad \forall v \in X^{\mathcal{N}}. \quad (3.6)$$

The corresponding “truth” output of interest is

$$s^{\mathcal{N}}(\boldsymbol{\mu}) \equiv l^{\text{out}}(u^{\mathcal{N}}(\boldsymbol{\mu}); \boldsymbol{\mu}) \equiv a(u^{\mathcal{N}}(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - f(v^*; \boldsymbol{\mu}). \quad (3.7)$$

By the appellation “truth”, we here understand that, for all $\boldsymbol{\mu} \in \mathcal{D}$, the FE solution and output errors are assumed smaller than some prescribed (problem dependent) tolerance. Moreover, for any given $\boldsymbol{\mu} \in \mathcal{D}$, the RB *a posteriori* error estimators (to which we return in Section 3.3) provide upper bounds only for the gap between the RB field, or output, and the corresponding “truth” FE field, or output, respectively. Hence, the RB fields and outputs are both built upon and estimated relative to the “truth” fields and outputs, respectively, which thus in effect serve as surrogates for the exact fields and outputs, respectively.

3.1.3 RB approximation

For $1 \leq N \leq N_{\max}$, the RB approximation space, X_N , will be the span of precomputed snapshots taken of $u^{\mathcal{N}}$ at N different points in \mathcal{D} . Specifically,

given a set of N wisely selected parameter vectors $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_N \in \mathcal{D}$, we define the corresponding RB space as

$$X_N = \text{span}\{u^{\mathcal{N}}(\boldsymbol{\mu}_1), \dots, u^{\mathcal{N}}(\boldsymbol{\mu}_N)\}. \quad (3.8)$$

We assume the $u^{\mathcal{N}}(\boldsymbol{\mu}_n)$, $1 \leq n \leq N$, to be linearly independent. The reduced basis problem now becomes: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u_N(\boldsymbol{\mu}) \in X_N$ such that

$$a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}), \quad \forall v \in X_N, \quad (3.9)$$

and evaluate the RB flux integral output

$$s_N(\boldsymbol{\mu}) \equiv l^{\text{out}}(u_N(\boldsymbol{\mu}); \boldsymbol{\mu}) = a(u_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - f(v^*; \boldsymbol{\mu}). \quad (3.10)$$

Starting from an initial (e.g. randomly chosen) $\boldsymbol{\mu}_1 \in \mathcal{D}$, the parameter vectors $\boldsymbol{\mu}_N$, $2 \leq N \leq N_{\text{max}}$, are chosen one at a time in a greedy manner. For each value of N , we evaluate the *a posteriori* (e.g. field energy norm) error estimator, $\Delta_{N-1}(\boldsymbol{\mu})$, for every $\boldsymbol{\mu}$ in a finite training sample, $\Xi_{\text{train}} \subset \mathcal{D}$. We then choose $\boldsymbol{\mu}_N$ equal to $\boldsymbol{\mu} \in \Xi_{\text{train}}$ that maximises $\Delta_{N-1}(\boldsymbol{\mu})$, and compute the corresponding snapshot $u^{\mathcal{N}}(\boldsymbol{\mu}_N)$ as the next RB basis function. The greedy parameter selection procedure was introduced in [22]. For a detailed description, see also [17, 20].

Note that we have assumed evaluation of the “truth” and RB flux integral outputs in (3.7) and (3.10), respectively, by using a flux lifting function $v^* \in V^*$, although we have not yet made any particular choice for v^* .

Under the assumption that $u^{\mathcal{N}}(\boldsymbol{\mu})$ varies smoothly with $\boldsymbol{\mu} \in \mathcal{D}$, we may expect that — for any $\boldsymbol{\mu} \in \mathcal{D}$ — very few RB basis functions will suffice in order to reproduce a very good RB approximation, $u_N(\boldsymbol{\mu})$, to the corresponding “truth” approximation, $u^{\mathcal{N}}(\boldsymbol{\mu})$, via the Galerkin formulation (3.9). Consequently, the corresponding RB output, $s_N(\boldsymbol{\mu})$, will be very close to the “truth” output, $s^{\mathcal{N}}(\boldsymbol{\mu})$. In particular, for a fixed level of accuracy relative to the *exact* field, $u(\boldsymbol{\mu})$, or output, $\tilde{s}(\boldsymbol{\mu})$, we expect that the number of required degrees of freedom associated with the RB approximations, N , is much lower than the number of required degrees of freedom associated with the “truth” approximations, \mathcal{N} , since the RB approximation space is specifically tailored to the problem at hand. As a result, only a small system of algebraic equations needs to be solved for each new given $\boldsymbol{\mu}$, once a sufficiently large RB space, X_N , is constructed.

Imperative to the efficiency of the RB method is a computational decoupling in offline (preprocessing) and online stages. The offline stage — which is performed only once — may be computationally very costly and includes the greedy

parameter selection process and the computation of the corresponding “truth” snapshots, i.e., the RB basis functions. The online stage — in which, given any new $\boldsymbol{\mu} \in \mathcal{D}$, the RB solution, $u_N(\boldsymbol{\mu})$, and RB output of interest, $s_N(\boldsymbol{\mu})$, are computed — is very fast. In particular, owing to the assumptions (3.2) on parametric affinity, the computational complexity of the RB online stage can be made independent of \mathcal{N} [17, 20].

Of interest within the RB context is the concept of a *compliant* problem. A problem is said to be compliant if, for all $\boldsymbol{\mu} \in \mathcal{D}$, *i*) the output functional $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ (more generally, in the case of $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ affine, the linear functional part of $l^{\text{out}}(\cdot; \boldsymbol{\mu})$) is equal to the right-hand-side $f(\cdot; \boldsymbol{\mu})$ of (3.1), and *ii*) $a(\cdot, \cdot; \boldsymbol{\mu})$ is symmetric. In the compliant case, the error in the RB output of interest is *equal* to the square of the energy-norm error of the primary field variable [17, 18, 20].

In our case, $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ as defined in (3.7) (or (3.10)) above is a non-compliant output functional since its linear functional part $a(\cdot, v^*; \boldsymbol{\mu})$ is, in general, different from the right-hand-side $f(\cdot; \boldsymbol{\mu})$ of (3.1). We make a comment in Section 4.4 on a very particular case in which $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ is, in fact, compliant.

3.2 Relevance of the flux lifting function

In Section 2, we saw that any two choices for v^* belonging to $(V^* \cap \tilde{X}^{\mathcal{N}})$ produced the same output $l^{\text{out}}(u^{\mathcal{N}})$ within a standard FE framework. This is of course a consequence of Galerkin orthogonality and, when compared to the RB space, X_N , the richness and generality of the FE space, $X^{\mathcal{N}}$. Within the RB framework however, the choice of flux lifting function does affect the numerical value of the output. To see this, let $v_1^*, v_2^* \in (V^* \cap \tilde{X}^{\mathcal{N}})$, and define $w^* \equiv v_1^* - v_2^*$. In order to denote the RB output for a particular $\boldsymbol{\mu} \in \mathcal{D}$, computed with a particular v^* , we write

$$s_N(\boldsymbol{\mu}; v^*) \equiv a(u_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - f(v^*; \boldsymbol{\mu}). \quad (3.11)$$

Hence, the outputs corresponding to v_1^* and v_2^* are given by

$$s_N(\boldsymbol{\mu}; v_1^*) = a(u_N(\boldsymbol{\mu}), v_1^*; \boldsymbol{\mu}) - f(v_1^*; \boldsymbol{\mu}), \quad (3.12)$$

$$s_N(\boldsymbol{\mu}; v_2^*) = a(u_N(\boldsymbol{\mu}), v_2^*; \boldsymbol{\mu}) - f(v_2^*; \boldsymbol{\mu}), \quad (3.13)$$

respectively. But then,

$$\begin{aligned} s_N(\boldsymbol{\mu}; v_1^*) - s_N(\boldsymbol{\mu}; v_2^*) &= a(u_N(\boldsymbol{\mu}), v_1^*; \boldsymbol{\mu}) - f(v_1^*; \boldsymbol{\mu}) \\ &\quad - (a(u_N(\boldsymbol{\mu}), v_2^*; \boldsymbol{\mu}) - f(v_2^*; \boldsymbol{\mu})) \\ &= a(u_N(\boldsymbol{\mu}), w^*; \boldsymbol{\mu}) - f(w^*; \boldsymbol{\mu}), \end{aligned} \quad (3.14)$$

which by (3.9) is equal to zero for all $w^* \in X_N$. However, there must clearly exist $w^* \in X^{\mathcal{N}}$ such that $a(u_N(\boldsymbol{\mu}), w^*; \boldsymbol{\mu}) - f(w^*; \boldsymbol{\mu})$ is *nonzero*. Otherwise, $u_N(\boldsymbol{\mu})$ would have been identical to $u^{\mathcal{N}}(\boldsymbol{\mu})$, which is provably not the case for a general $\boldsymbol{\mu} \in \mathcal{D}$. In conclusion, the two evaluations $s_N(\boldsymbol{\mu}, v_1^*)$ and $s_N(\boldsymbol{\mu}, v_2^*)$ are not in general equivalent (unless $v_1^* - v_2^*$ happens to belong to X_N). Naturally, this raises the question of which v^* we should choose within the RB framework.

3.3 *A posteriori* error estimation

Before we proceed to our actual choices for “good” RB flux lifting functions, we shall consider the *a posteriori* error upper bound associated with the output $s_N(\boldsymbol{\mu})$. To arrive at such a bound, we first require a bound for the error in the field variable. We assume that we have available a lower bound $\alpha_{\text{LB}}(\boldsymbol{\mu}) > 0$ for the coercivity constant of $a(\cdot, \cdot; \boldsymbol{\mu})$ over $X^{\mathcal{N}}$ with respect to the X -norm defined in (3.5). Specifically, for all $\boldsymbol{\mu} \in \mathcal{D}$,

$$\alpha_{\text{LB}}(\boldsymbol{\mu}) \leq \alpha(\boldsymbol{\mu}) = \inf_{v \in X^{\mathcal{N}}} \frac{a(v, v; \boldsymbol{\mu})}{\|v\|_X^2}. \quad (3.15)$$

We also define the residual

$$r_N(v; \boldsymbol{\mu}) \equiv f(v; \boldsymbol{\mu}) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) \quad (3.16)$$

for all $v \in X^{\mathcal{N}}$. In particular, with $e_N(\boldsymbol{\mu}) = u^{\mathcal{N}}(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})$, we have

$$a(e_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = r_N(v; \boldsymbol{\mu}), \quad \forall v \in X^{\mathcal{N}}. \quad (3.17)$$

Hence, with $\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ that satisfies

$$a(\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu}), v; \bar{\boldsymbol{\mu}}) = r_N(v; \boldsymbol{\mu}), \quad \forall v \in X^{\mathcal{N}}, \quad (3.18)$$

we may write

$$\begin{aligned} a(e_N(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}); \boldsymbol{\mu}) &= a(\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}); \bar{\boldsymbol{\mu}}) \\ &\leq \|\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu})\|_X \|e_N(\boldsymbol{\mu})\|_X \\ &\leq \|\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu})\|_X \frac{\|e_N(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}}{(\alpha_{\text{LB}}(\boldsymbol{\mu}))^{1/2}}, \end{aligned} \quad (3.19)$$

Evaluation of Flux Integral Outputs

where we have used $v = e_N(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ in (3.17) and (3.18), the Cauchy-Schwarz inequality and the definition (3.15) of the coercivity lower bound. Hence,

$$\|u^{\mathcal{N}}(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \leq \frac{\|\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu})\|_X}{(\alpha_{\text{LB}}(\boldsymbol{\mu}))^{1/2}} \equiv \Delta_N(\boldsymbol{\mu}), \quad (3.20)$$

where $\|\hat{e}_N^{\mathcal{N}}(\boldsymbol{\mu})\|_X = \sup_{v \in X^{\mathcal{N}}} (r_N(v; \boldsymbol{\mu}) / \|v\|_X)$ is the dual norm of the residual. Again, due to the affinity assumptions (3.2), an efficient offline-online computational approach for $\Delta_N(\boldsymbol{\mu})$ can be developed. For a detailed derivation of (3.20) and the corresponding computational procedure, see [17, 18, 20].

Now, we let $\tilde{X}^{\mathcal{N}} \supset X^{\mathcal{N}}$ be a discrete FE space identical to $X^{\mathcal{N}}$ except for the restriction on its members vanishing on Γ_0^{D} . Then we let $\psi^{\mathcal{N}}(\boldsymbol{\mu}) \in (V^* \cap \tilde{X}^{\mathcal{N}})$ be the solution of the problem

$$a(v, \psi^{\mathcal{N}}(\boldsymbol{\mu}); \boldsymbol{\mu}) = 0, \quad \forall v \in X^{\mathcal{N}}. \quad (3.21)$$

Note that since $\psi^{\mathcal{N}}(\boldsymbol{\mu}) \in (V^* \cap \tilde{X}^{\mathcal{N}})$, we impose the essential Dirichlet condition $\psi^{\mathcal{N}}(\boldsymbol{\mu})|_{\Gamma_0^{\text{D}}} = 1$.

Next, if we choose any $v^* \in (V^* \cap \tilde{X}^{\mathcal{N}})$, our “truth output” — to which the RB output will be compared — is given from (3.7) as

$$s^{\mathcal{N}}(\boldsymbol{\mu}) = a(u^{\mathcal{N}}(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - f(v^*; \boldsymbol{\mu}), \quad (3.22)$$

and we obtain the error estimate

$$\begin{aligned} |s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; v^*)| &= |a(u^{\mathcal{N}}(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - a(u_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu})| \\ &= |a(e_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu})| \\ &= |a(e_N(\boldsymbol{\mu}), v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &\leq \|e_N(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}, \end{aligned} \quad (3.23)$$

by using (3.21) with $e_N(\boldsymbol{\mu}) \in X^{\mathcal{N}}$, and the Cauchy-Schwarz inequality. Thus, $|s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; v^*)| \leq \Delta_N(\boldsymbol{\mu}) \|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$, and a good v^* is also a good approximation to $\psi^{\mathcal{N}}(\boldsymbol{\mu})$, making the term $\|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$ small.

To bound the term $\|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$, we view v^* as an approximation to $\psi^{\mathcal{N}}$, and define the residual $r_{v^*}(v; \boldsymbol{\mu}) \equiv -a(v, v^*; \boldsymbol{\mu})$. Analogously to (3.20), we have

$$\|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \leq \frac{\|\hat{e}_{v^*}^{\mathcal{N}}(\boldsymbol{\mu})\|_X}{(\alpha_{\text{LB}}(\boldsymbol{\mu}))^{1/2}} \equiv \Delta_{v^*}(\boldsymbol{\mu}), \quad (3.24)$$

where $\hat{e}_{v^*}^{\mathcal{N}}(\boldsymbol{\mu})$ belongs to $X^{\mathcal{N}}$ and solves $a(\hat{e}_{v^*}^{\mathcal{N}}(\boldsymbol{\mu}), v; \bar{\boldsymbol{\mu}}) = r_{v^*}(v; \boldsymbol{\mu})$ for all $v \in X^{\mathcal{N}}$. We thus arrive at

$$|s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; v^*)| \leq \Delta_N(\boldsymbol{\mu})\Delta_{v^*}(\boldsymbol{\mu}) \equiv \Delta_{N,v^*}^{\text{out}}(\boldsymbol{\mu}) \quad (3.25)$$

as an upper bound for our output of interest.

Note that if we write $\psi^{\mathcal{N}}(\boldsymbol{\mu}) = \psi^{\mathcal{N},0}(\boldsymbol{\mu}) + \psi^{\text{D}}$, where $\psi^{\text{D}} \in (V^* \cap \tilde{X}^{\mathcal{N}})$ is some chosen Dirichlet lift, we can write (3.21) as: Given $\boldsymbol{\mu} \in \mathcal{D}$, find $\psi^{\mathcal{N},0}(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ such that

$$a(v, \psi^{\mathcal{N},0}(\boldsymbol{\mu}); \boldsymbol{\mu}) = -a(v, \psi^{\text{D}}; \boldsymbol{\mu}), \quad \forall v \in X^{\mathcal{N}}. \quad (3.26)$$

Thus, if we choose ψ^{D} equal to v^* , (3.26) is in fact the dual problem corresponding to the *primal* problem (3.6) with the output functional $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ given in (3.7), since $a(\cdot, v^*; \boldsymbol{\mu})$ is the linear functional part of $l^{\text{out}}(\cdot; \boldsymbol{\mu})$. We elaborate on this in Section 3.5.

3.4 “Good” flux lifting functions

We must keep two things in mind when choosing the flux lifting function v^* . Firstly, it is important that the term $\|v^* - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$ in the estimate (3.23) is small. Secondly, we must make sure that the computational cost associated with the computation of v^* is small in the RB online stage.

Note that actually solving (3.21) for every new $\boldsymbol{\mu}$ and setting $v^* = \psi^{\mathcal{N}}(\boldsymbol{\mu})$ will result in a zero error in the RB output, but obviously also in an unaffordable “truth FE complexity” online computational cost.

We next consider two alternative choices of “good” v^* ’s which both meet the two requirements mentioned above.

3.4.1 Coarse finite element approximation

Our first choice is to construct a coarse finite element approximation to $\psi^{\mathcal{N}}(\boldsymbol{\mu})$. That is to say, we first find $\psi^{\mathcal{M}}(\boldsymbol{\mu}) \in (V^* \cap \tilde{X}^{\mathcal{M}}) \subset (V^* \cap \tilde{X}^{\mathcal{N}})$ such that

$$a(v, \psi^{\mathcal{M}}(\boldsymbol{\mu}); \boldsymbol{\mu}) = 0, \quad \forall v \in X^{\mathcal{M}}, \quad (3.27)$$

where the coarse FE space $X^{\mathcal{M}} \subset X^{\mathcal{N}}$ has dimension $\mathcal{M} \ll \mathcal{N}$, and $\tilde{X}^{\mathcal{M}}$ is equal to $X^{\mathcal{M}}$ except for the restriction on its members vanishing on Γ_0^{D} . We then choose $v^* = \psi^{\mathcal{M}}(\boldsymbol{\mu})$ as the flux lifting function. In particular, \mathcal{M} should here be chosen small enough that it is affordable to compute $\psi^{\mathcal{M}}(\boldsymbol{\mu})$ in the RB online stage, without compromising the rapid online RB output evaluation.

Evaluation of Flux Integral Outputs

Now, to bound the term $\|\psi^{\mathcal{M}}(\boldsymbol{\mu}) - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$, we use the result (3.24) to arrive at $\|\psi^{\mathcal{M}}(\boldsymbol{\mu}) - \psi^{\mathcal{N}}(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \leq \Delta_{\mathcal{M}}(\boldsymbol{\mu})$ (we here use \mathcal{M} to indicate that the v^* in (3.24) is now a coarse FE approximation to $\psi^{\mathcal{N}}(\boldsymbol{\mu})$). Then, we define $\Delta_{N,\mathcal{M}}^{\text{out}}(\boldsymbol{\mu}) \equiv \Delta_N(\boldsymbol{\mu})\Delta_{\mathcal{M}}(\boldsymbol{\mu})$, and conclude that for all $\boldsymbol{\mu} \in \mathcal{D}$,

$$|s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; \psi^{\mathcal{M}}(\boldsymbol{\mu}))| \leq \Delta_{N,\mathcal{M}}^{\text{out}}(\boldsymbol{\mu}). \quad (3.28)$$

3.4.2 Reference parameter approximation

Alternatively, we may take $v^* = \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}})$ as our approximation of $\psi^{\mathcal{N}}(\boldsymbol{\mu})$ for any $\boldsymbol{\mu} \in \mathcal{D}$. Since $v^* = \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}})$ is the solution to (3.21) for the preselected reference parameter, it can be precomputed in the RB offline stage and then reused every time we evaluate the RB output of interest, without any additional RB online cost.

From (3.24), an upper bound for the term $\|\psi^{\mathcal{N}}(\boldsymbol{\mu}) - \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}})\|_{\boldsymbol{\mu}}$ is given by $\Delta_{\bar{\boldsymbol{\mu}}}(\boldsymbol{\mu})$ (where we substitute $\bar{\boldsymbol{\mu}}$ for v^* to remember our particular choice for v^*). We thus get

$$|s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}}))| \leq \Delta_N(\boldsymbol{\mu})\Delta_{\bar{\boldsymbol{\mu}}}(\boldsymbol{\mu}) \equiv \Delta_{N,\bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu}) \quad (3.29)$$

as an upper bound for the error in the RB output of interest.

3.5 Primal-dual RB approximation

Evidently, one way to approximate $\psi^{\mathcal{N}}(\boldsymbol{\mu})$ is by way of a reduced basis approximation $\psi_M(\boldsymbol{\mu})$. The RB problem corresponding to (3.26) (and (3.21)) reads: Find $\psi_M^0(\boldsymbol{\mu}) \in X_M^{\text{du}}$ such that

$$a(v, \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) = -a(v, \psi^{\text{D}}; \boldsymbol{\mu}), \quad \forall v \in X_M^{\text{du}}, \quad (3.30)$$

and set $\psi_M(\boldsymbol{\mu}) = \psi_M^0(\boldsymbol{\mu}) + \psi^{\text{D}}$. Here, X_M^{du} denotes the RB dual approximation space, given by

$$X_M^{\text{du}} = \text{span}\{\psi^{\mathcal{N}}(\boldsymbol{\mu}_m) - \psi^{\text{D}}\}_{m=1}^M, \quad (3.31)$$

where the $\psi^{\mathcal{N}}(\boldsymbol{\mu}_m)$ are snapshots taken of $\psi^{\mathcal{N}}$ at M different points in parameter space. In fact, the formulation of the two problems (3.9) and (3.30), together with the output of interest given in (3.10), correspond to a standard RB primal-dual formulation, which is the standard way of speeding up the convergence of RB solutions to general non-compliant problems [18, 20]. Let us spend a few lines elaborating on this.

Paper 1

First, we choose a $v^* \in (V^* \cap \tilde{X}^{\mathcal{N}})$ and let $\psi^{\text{D}} = v^*$. Hence, the right-hand-side of (3.30) is precisely the linear functional part of the output functional (with a minus sign), and v^* is the Dirichlet lifting function for the dual problem. The standard “dual-corrected” RB output reads

$$\hat{s}_{M,N}(\boldsymbol{\mu}) \equiv s_N(\boldsymbol{\mu}; \psi^{\text{D}}) - r_N(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}), \quad (3.32)$$

where $r_N(v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu})$ is the primal residual and $\psi_M^0(\boldsymbol{\mu})$ is the homogeneous part of the solution to (3.30). In the below expression, we drop the argument $\boldsymbol{\mu}$ of functions in all intermediate steps for brevity. With $e_N(\boldsymbol{\mu}) = u^{\mathcal{N}}(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})$, we see that

$$\begin{aligned} |s^{\mathcal{N}}(\boldsymbol{\mu}) - \hat{s}_{M,N}(\boldsymbol{\mu})| &= |s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; \psi^{\text{D}}) + r_N(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N, \psi^{\text{D}}; \boldsymbol{\mu}) + r_N(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N, \psi^{\mathcal{N},0}; \boldsymbol{\mu}) - r_N(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N, \psi^{\mathcal{N},0}; \boldsymbol{\mu}) + a(u_N, \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) - f(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N, \psi^{\mathcal{N},0}; \boldsymbol{\mu}) + a(u_N, \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) - a(u^{\mathcal{N}}, \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N, \psi^{\mathcal{N},0} - \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &\leq \|e_N(\boldsymbol{\mu})\|_{\boldsymbol{\mu}} \|\psi^{\mathcal{N},0}(\boldsymbol{\mu}) - \psi_M^0(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}. \end{aligned} \quad (3.33)$$

In the two first steps, we here use the expression (3.32) for $\hat{s}_{M,N}$ and then arbitrariness (up to functions in $(V^* \cap \tilde{X}^{\mathcal{N}})$) of the flux lifting function for the “truth” output (we can thus write $s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; \psi^{\text{D}}) = a(e_N(\boldsymbol{\mu}), \psi^{\text{D}}; \boldsymbol{\mu})$). In the third and fourth steps, we invoke (3.26) with $v = -e_N(\boldsymbol{\mu})$ and then (3.16) with $v = \psi_M^0$. Next, we use the definition of the “truth” primal problem, then linearity and lastly the Cauchy-Schwarz inequality. Hence, if we solve the RB primal and dual problems in parallel with $M \approx N$, we get a “quadratic” effect in the convergence of the output of interest.

Next, it is straightforward to deduce that $\hat{s}_{M,N}(\boldsymbol{\mu}) = s_N(\boldsymbol{\mu}; \psi_M(\boldsymbol{\mu}))$, i.e., that these two output evaluations are equivalent. We start with the expression (3.32), and then appeal to the (bi)linearity of $a(\cdot, \cdot; \boldsymbol{\mu})$ and $f(\cdot; \boldsymbol{\mu})$. Again, we drop the $\boldsymbol{\mu}$ -dependence of functions for typesetting convenience:

$$\begin{aligned} \hat{s}_{M,N}(\boldsymbol{\mu}) &= s_N(\boldsymbol{\mu}; \psi^{\text{D}}) - r_N(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) \\ &= a(u_N, \psi^{\text{D}}; \boldsymbol{\mu}) + a(u_N, \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) - f(\psi^{\text{D}}; \boldsymbol{\mu}) - f(\psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) \\ &= a(u_N, \psi^{\text{D}} + \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) - f(\psi^{\text{D}} + \psi_M^0(\boldsymbol{\mu}); \boldsymbol{\mu}) \\ &= s_N(\boldsymbol{\mu}; \psi_M). \end{aligned} \quad (3.34)$$

Evaluation of Flux Integral Outputs

In other words, the standard dual-corrected output with $v^* = \psi^D$ produces the same result as the non-corrected output with $v^* = \psi_M(\boldsymbol{\mu}) = \psi_M^0(\boldsymbol{\mu}) + \psi^D$. Thus, for flux integral outputs, the standard RB primal-dual approximation framework may be viewed as a technique for improving upon any initial choice $v^* = \psi^D$.

Up to this point, we have only considered a single output of interest. Surely, it could in a practical application be desirable to evaluate several outputs of interest, all being functionals of the solution of the same underlying PDE. For example, we might want the integral of the flux through J entire disjoint sections $\Gamma_0^D, \Gamma_1^D, \dots, \Gamma_{J-1}^D$ of the boundary, resulting in J different output functionals and thus in turn J different dual problems. Of course, in the case of multiple outputs, no more than one can be compliant.

When J is small, solving the primal and dual problems in parallel with (say) $M \approx N$ basis functions may drastically reduce the RB output error(s) and error bound(s). However, for many outputs (large J), online computation of the solution to every corresponding dual problem (when $M \approx N$) may be impracticable — $\mathcal{O}(N^3)$ and $\mathcal{O}(M^3)$ operations are required for direct computation of the solutions to the primal and dual RB problems, respectively — and we are thus forced to trade numerical accuracy for computational speed. In this situation, choosing good flux lifting functions seems important.

On the other hand, if we do proceed with RB approximations to the solution(s) to the dual problem(s) as well, making good choices for the dual Dirichlet liftings would surely be of interest (obviously, $\psi^D = \psi^N(\boldsymbol{\mu})$ would be the optimal, though an impractical, choice).

3.6 Computational approach for output evaluation

In the RB online stage, and without regard to our particular choice for v^* , we need to compute

$$s_N(\boldsymbol{\mu}) = l^{\text{out}}(u_N(\boldsymbol{\mu}); \boldsymbol{\mu}) = a(u_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}) - f(v^*; \boldsymbol{\mu}), \quad (3.35)$$

once the RB solution is obtained. Note that also when we pursue a primal-dual approximation, we are still left with an evaluation on this form, due to the result (3.34).

Since, for each $\boldsymbol{\mu} \in \mathcal{D}$, $a(\cdot, \cdot; \boldsymbol{\mu})$ and $f(\cdot; \boldsymbol{\mu})$ are, by assumption, parametrically affine, $l^{\text{out}}(\cdot; \boldsymbol{\mu})$ will also be parametrically affine. Hence, we can compute the RB output at an additional computational cost of $\mathcal{O}(N)$ operations. To see this, we write $a(v, w; \boldsymbol{\mu})$ and $f(v; \boldsymbol{\mu})$ in their parametrically affine expansions

(3.2) as

$$a(v, w; \boldsymbol{\mu}) = \sum_{q=1}^{Q_a} a^q(v, w) \Theta_a^q(\boldsymbol{\mu}), \quad f(v; \boldsymbol{\mu}) = \sum_{q=1}^{Q_f} f^q(v) \Theta_f^q(\boldsymbol{\mu}), \quad (3.36)$$

for any $v, w \in X$. With $u_N(\boldsymbol{\mu}) = \sum_{n=1}^N u_{N,n}(\boldsymbol{\mu}) \zeta_n$, where the ζ_n are the orthogonalised RB basis functions (in order to get a reduced system of equations that is well conditioned, the basis functions $u^N(\boldsymbol{\mu}_n)$, $1 \leq n \leq N$, are orthonormalised with respect to the norm $\|\cdot\|_X$, cf. [20]) and $u_{N,n}(\boldsymbol{\mu})$ are the RB solution coefficients, we get

$$\begin{aligned} l^{\text{out}}(u_N(\boldsymbol{\mu}); \boldsymbol{\mu}) &= \sum_{q=1}^{Q_a} a^q(u_N(\boldsymbol{\mu}), v^*) \Theta_a^q(\boldsymbol{\mu}) - \sum_{q=1}^{Q_f} f^q(v^*) \Theta_f^q(\boldsymbol{\mu}) \\ &= \sum_{n=1}^N u_{N,n}(\boldsymbol{\mu}) \sum_{q=1}^{Q_a} a^q(\zeta_n, v^*) \Theta_a^q(\boldsymbol{\mu}) - \sum_{q=1}^{Q_f} f^q(v^*) \Theta_f^q(\boldsymbol{\mu}), \end{aligned} \quad (3.37)$$

which is a $Q_a N + Q_f$ operations summation, assuming that the values $a^q(\zeta_n, v^*)$, $1 \leq q \leq Q_a$, and $f^q(v^*)$, $1 \leq q \leq Q_f$ are precomputed in the RB offline stage, and that the function values $\Theta_a^q(\boldsymbol{\mu})$ and $\Theta_f^q(\boldsymbol{\mu})$ are readily computable. Note that we have here assumed for simplicity that v^* is $\boldsymbol{\mu}$ -independent (according to, for example, the reference parameter approximation discussed in Section 3.4.2).

Thus far, we have assumed that a and f are parametrically affine. In the more general case of a non-affine problem, it is possible to construct parametrically affine expansions that are good *approximations* of $a(\cdot, \cdot; \boldsymbol{\mu})$ and $f(\cdot; \boldsymbol{\mu})$ for any $\boldsymbol{\mu} \in \mathcal{D}$ by invoking the *empirical interpolation* method [3, 9, 14]. In this case, the additional computational cost for output evaluation is still only $\mathcal{O}(N)$ in the RB online stage, but the RB problem solved is slightly modified.

4 An Illustrative Example

4.1 Problem formulation

We consider the electrostatic potential, u , inside a square domain $\Omega = (0, 3) \times (0, 3)$ which contains an “insulating” square anomaly, Ω_{an} , with edges of length 0.6, as depicted in Figure 3.

Attached to the boundary of Ω , $\partial\Omega$, are four electrodes of unity length, centred on each of the edges of Ω . The electrodes constitute the Dirichlet

Evaluation of Flux Integral Outputs

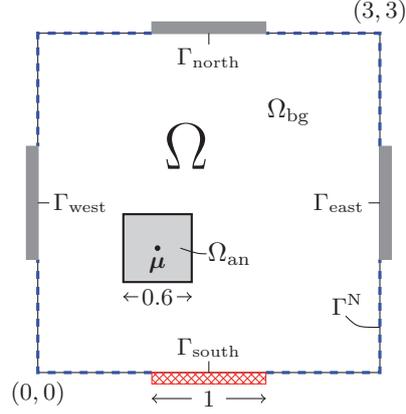


Figure 3: Physical domain Ω with an electrode attached to each edge.

boundary $\Gamma^D \equiv \Gamma_{\text{south}} \cup \Gamma_{\text{north}} \cup \Gamma_{\text{east}} \cup \Gamma_{\text{west}}$, on which the potential is prescribed as

$$u = \begin{cases} 1, & \text{on } \Gamma_{\text{south}}, \\ 0, & \text{on } \Gamma_{\text{north}} \cup \Gamma_{\text{east}} \cup \Gamma_{\text{west}}. \end{cases} \quad (4.1)$$

On the Neumann boundary, $\Gamma^N \equiv \partial\Omega \setminus \Gamma^D$, we assume electrostatic insulation, i.e.

$$\frac{\partial u}{\partial n} = 0, \quad \text{on } \Gamma^N. \quad (4.2)$$

We define the “background material” as $\Omega_{\text{bg}} \equiv \Omega \setminus \bar{\Omega}_{\text{an}}$. The electric permittivity, ϵ , inside Ω is given as

$$\epsilon \equiv \begin{cases} \epsilon_{\text{bg}} \equiv 1, & \text{in } \Omega_{\text{bg}}, \\ \epsilon_{\text{an}} \equiv 0.1, & \text{in } \Omega_{\text{an}}. \end{cases} \quad (4.3)$$

Finally, in $\Omega_{\text{bg}} \cup \Omega_{\text{an}}$, the electrostatic potential is governed by the Laplace equation,

$$-\nabla^2 u = 0. \quad (4.4)$$

Our problem is parametrised by the parameter vector $\boldsymbol{\mu} \equiv (\mu_1, \mu_2) \in \mathcal{D}$, which determines the position of the centre of Ω_{an} . Here, $\mathcal{D} \subset \Omega$ is our parameter

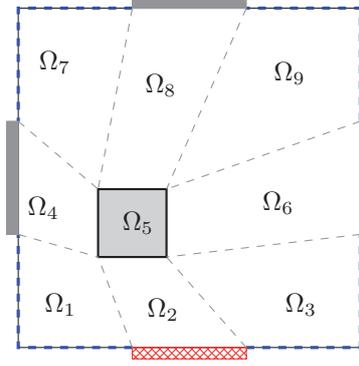


Figure 4: Decomposition of the physical domain into nine (deformed square) spectral elements.

domain, defined as $\mathcal{D} \equiv [1, 2] \times [1, 2]$. Hence, the shape of Ω_{bg} and the position of Ω_{an} depend upon $\boldsymbol{\mu}$. For typesetting convenience however, we do not explicitly denote these dependencies in formulas.

Our output of interest is the accumulated charge on the eastern electrode, i.e., the capacitance corresponding to the “south-east” pair of electrodes, given by the flux integral

$$\tilde{s}(\boldsymbol{\mu}) \equiv -\epsilon_{\text{bg}} \int_{\Gamma_{\text{east}}} \frac{\partial u(\boldsymbol{\mu})}{\partial n} ds. \quad (4.5)$$

4.2 RB treatment

4.2.1 Parametrised weak form

Let $u^{\text{D}} \in H^1$ be a lifting of the Dirichlet data (4.1), and write $u(\boldsymbol{\mu}) = u^0(\boldsymbol{\mu}) + u^{\text{D}}$. With the boundary conditions (4.1) and (4.2), together with the assumption of flux continuity on the interior boundary and global continuity of the solution, the parametric weak form of our problem reads: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u^0(\boldsymbol{\mu}) \in X$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = 0, \quad \forall v \in X, \quad (4.6)$$

Evaluation of Flux Integral Outputs

where $u(\boldsymbol{\mu}) = u^0(\boldsymbol{\mu}) + u^D$,

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = \epsilon_{\text{bg}} \int_{\Omega_{\text{bg}}} \nabla u(\boldsymbol{\mu}) \cdot \nabla v \, d\Omega + \epsilon_{\text{an}} \int_{\Omega_{\text{an}}} \nabla u(\boldsymbol{\mu}) \cdot \nabla v \, d\Omega \quad (4.7)$$

and $X = \{v \in H^1 : v|_{\Gamma^D} = 0\}$.

In order to treat our problem numerically, we decompose our domain into nine subdomains, as depicted in Figure 4. Next, we introduce a reference domain $\hat{\Omega} \equiv (-1, 1) \times (-1, 1)$ and standard transfinite mappings [8] $\mathcal{F}_i : \Omega_i \rightarrow \hat{\Omega}$, $1 \leq i \leq 9$. We may then write our bilinear form, a , in terms of the reference variables (ξ, η) on the reference domain as

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = \sum_{i=1}^9 \epsilon_i \int_{\hat{\Omega}} (\hat{\nabla} \hat{u}_i(\boldsymbol{\mu}))^T G_i(\boldsymbol{\mu}) \hat{\nabla} \hat{v}_i \, d\hat{\Omega}, \quad (4.8)$$

where $\hat{\nabla}$ is the gradient operator in the reference variables and, for $1 \leq i \leq 9$, ϵ_i denotes the electric permittivity in Ω_i , $\hat{v}_i(\xi, \eta) \equiv v(x, y)|_{\Omega_i} \circ \mathcal{F}_i$, and G_i is a parametrically and spatially dependent 2×2 symmetric positive definite matrix comprising the geometrical factors induced by \mathcal{F}_i .

4.2.2 “Truth” spectral element approximation

In order to construct the snapshots upon which to build the RB approximations, we define the “truth” discretisation of (4.6). To this end, we shall employ a standard spectral element method based on high-order polynomials [15].

As our discrete “truth” approximation space, we define

$$X^{\mathcal{N}} \equiv \{v \in H^1(\Omega) : v|_{\Gamma^D} = 0, \hat{v}_i \in \mathbb{P}_P(\hat{\Omega}), 1 \leq i \leq 9\}, \quad (4.9)$$

where $\mathbb{P}_P(\hat{\Omega})$ denotes the space of polynomials of degree P in each direction over $\hat{\Omega}$. With $\mathcal{N} = \dim(X^{\mathcal{N}})$, we note that $\mathcal{N} \sim P^2$. For the Dirichlet and flux lifting functions, we shall also use (mapped) polynomials. We thus require the spaces

$$\tilde{X}_{\text{east}}^{\mathcal{N}} \equiv \{v \in H^1(\Omega) : v|_{\Gamma^D \setminus \Gamma_{\text{east}}} = 0, \hat{v}_i \in \mathbb{P}_P(\hat{\Omega}), 1 \leq i \leq 9\}, \quad (4.10)$$

$$\tilde{X}_{\text{south}}^{\mathcal{N}} \equiv \{v \in H^1(\Omega) : v|_{\Gamma^D \setminus \Gamma_{\text{south}}} = 0, \hat{v}_i \in \mathbb{P}_P(\hat{\Omega}), 1 \leq i \leq 9\}, \quad (4.11)$$

and we also define

$$V^* \equiv \{v \in H^1 : v|_{\Gamma^D \setminus \Gamma_{\text{east}}} = 0, v|_{\Gamma_{\text{east}}} = 1\}. \quad (4.12)$$

We now state the “truth” discretisation of (4.6) as: For any $\boldsymbol{\mu} \in \mathcal{D}$, find $u^{\mathcal{N},0}(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ such that

$$a(u^{\mathcal{N}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = 0, \quad \forall v \in X^{\mathcal{N}}, \quad (4.13)$$

where $u^{\mathcal{N}}(\boldsymbol{\mu}) = u^{\mathcal{N},0}(\boldsymbol{\mu}) + u^{\text{D}}$ and $u^{\text{D}} \in \tilde{X}_{\text{south}}^{\mathcal{N}}$ is some chosen ($\boldsymbol{\mu}$ -independent) lifting of the Dirichlet data (4.1). The corresponding “truth” output of interest is

$$s^{\mathcal{N}}(\boldsymbol{\mu}) = l^{\text{out}}(u^{\mathcal{N}}(\boldsymbol{\mu}); \boldsymbol{\mu}) = a(u^{\mathcal{N}}(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}), \quad (4.14)$$

In (4.14), we have omitted the minus sign (that appeared in (4.5) since the capacitance is positive by definition). Also note that for the “truth” problem, any two $v^* \in (V^* \cap \tilde{X}_{\text{east}}^{\mathcal{N}})$ will produce the same numerical output, as we saw in Section 2.

As basis functions for the three discrete spaces above, and as shape functions for the “truth” approximations, we use the Lagrange polynomials over the $(P+1)^2$ tensorised GLL nodes (on $\hat{\Omega}$) [5]. Specifically, we have made the choice $P = 35$.

Due to the change from Neumann to Dirichlet boundary conditions at the electrode edges, u exhibits known singularities [2, 11] which will limit the convergence of the high order polynomial approximation. Although there exist techniques to improve the finite element convergence rate when singularities are present, (e.g. using an adaptive h - p finite element method [2]) we restrict ourselves here to a standard spectral element approximation based on high order polynomials since our focus in this paper is the accuracy of the RB approximation relative to the “truth” approximation.

4.2.3 Reduced basis approximation

The RB approximation spaces are defined as

$$X_N \equiv \text{span}\{u^{\mathcal{N},0}(\boldsymbol{\mu}_n)\}_{n=1}^N, \quad (4.15)$$

for $1 \leq N \leq N_{\text{max}}$. If we move the term comprising the Dirichlet lifting to the right-hand-side, the RB problem may be written as: For any $\boldsymbol{\mu} \in \mathcal{D}$, find $u_N^0(\boldsymbol{\mu}) \in X_N$ such that

$$a(u_N^0(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = -a(u^{\text{D}}, v; \boldsymbol{\mu}), \quad \forall v \in X_N, \quad (4.16)$$

Evaluation of Flux Integral Outputs

and set $u_N(\boldsymbol{\mu}) = u_N^0(\boldsymbol{\mu}) + u^D$. We then evaluate our output of interest as

$$s_N(\boldsymbol{\mu}; v^*) \equiv a(u_N(\boldsymbol{\mu}), v^*; \boldsymbol{\mu}), \quad (4.17)$$

where we choose $v^* \in (V^* \cap \tilde{X}_{\text{cast}}^N)$.

In order to construct X_N , $1 \leq N \leq N_{\text{max}}$, we use the greedy parameter selection process outlined in Section 3.1.3. We start by choosing $\boldsymbol{\mu}_1$ randomly from \mathcal{D} and then compute the corresponding first snapshot $u^{N,0}(\boldsymbol{\mu}_1)$. Next, for $2 \leq N \leq N_{\text{max}}$, we choose $\boldsymbol{\mu}_N$ from a training sample $\Xi_{\text{train}} \subset \mathcal{D}$ in a greedy manner based on *a posteriori* upper bounds $\Delta_{N-1}(\boldsymbol{\mu})$ — computed for every $\boldsymbol{\mu} \in \Xi_{\text{train}}$ — for the energy-norm errors $\|u^{N,0}(\boldsymbol{\mu}) - u_{N-1}^0(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}$, and then compute the next snapshot as $u^{N,0}(\boldsymbol{\mu}_N)$. (A detailed description of the procedure can be found in [17, 20].) Here, we have chosen Ξ_{train} as an equidistant “grid” of 225 points in \mathcal{D} .

Unfortunately, the elements of the matrices G_k do not permit a parametrically affine expansion of a , as assumed in (3.2). For this reason, our computations do not immediately decouple into the very desirable offline and online stages. However, as commented in the previous section, the empirical interpolation method provides means to this end [3, 9, 14]. In fact, we can make the empirical interpolation error negligible if we make sure to include enough terms in the approximate affine expansion of a . If the number of required terms is not too large, we can at the same time use an efficient offline-online computational approach. In the numerical tests that follow, however, we have chosen to not use an offline-online decoupling approach, since this is not critical for our conclusions.

4.2.4 *A posteriori* error estimation: a coercivity lower bound

If we were to use an offline-online decoupling approach for our particular problem, we should also include additional terms in the *a posteriori* error estimators in order to account for the empirical interpolation error [3, 16]. However, the standard estimators from Section 3 are still valid under the assumption of a negligible interpolation error because the additional terms will vanish as the interpolation error goes to zero. Hence, if we for any $\boldsymbol{\mu} \in \mathcal{D}$ can establish a coercivity lower bound $\alpha_{\text{LB}}(\boldsymbol{\mu})$, we can compute an energy-norm error bound $\Delta_N(\boldsymbol{\mu})$ and output error bounds $\Delta_{N,\mathcal{M}}^{\text{out}}(\boldsymbol{\mu})$ and $\Delta_{N,\bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu})$ as described in Sections 3.3, 3.4.1 and 3.4.2. In fact, if we for $1 \leq k \leq 9$ let $\sigma_k(\boldsymbol{\mu})$ denote the set of eigenvalues of the (symmetric and positive definite) matrix $G_k(\boldsymbol{\mu})$ of geometrical

factors, and define

$$\lambda^-(\boldsymbol{\mu}) \equiv \min_{\substack{(\xi, \eta) \in \hat{\Omega} \\ 1 \leq k \leq 9}} \sigma_k(\xi, \eta; \boldsymbol{\mu}), \quad \lambda^+(\boldsymbol{\mu}) \equiv \max_{\substack{(\xi, \eta) \in \hat{\Omega} \\ 1 \leq k \leq 9}} \sigma_k(\xi, \eta; \boldsymbol{\mu}), \quad (4.18)$$

it can be shown (c.f. [7, 13]) that a coercivity lower bound for our particular problem is given by

$$\alpha_{\text{LB}}(\boldsymbol{\mu}) = \frac{\lambda^-(\boldsymbol{\mu})}{\lambda^+(\bar{\boldsymbol{\mu}})}. \quad (4.19)$$

Our particular choice for the reference parameter vector is $\bar{\boldsymbol{\mu}} = (1.5, 1.5)$, i.e., the centre of \mathcal{D} . We make a remark here that we in practice realise the maximum and minimum of the $\sigma_k(\cdot, \cdot; \boldsymbol{\mu})$ over the tensorised GLL nodes. An efficient offline–online decomposition procedure for the coercivity lower bound — *the successive constraint method* — can be found in [20].

4.3 Numerical results

Here, we present results for the RB output error, defined as

$$e_N^{\text{out}}(\boldsymbol{\mu}; v^*) \equiv |s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; v^*)|, \quad (4.20)$$

and the RB output error bound $\Delta_{N, v^*}^{\text{out}}(\boldsymbol{\mu})$. For $v^* \in (V^* \cap \tilde{X}_{\text{east}}^{\mathcal{N}})$, we shall make use of three different functions: The reference parameter approximation $\psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}})$ discussed in Section 3.4.2 with $\bar{\boldsymbol{\mu}} = (1.5, 1.5)$, the coarse finite element approximation $\psi^{\mathcal{M}_2}(\boldsymbol{\mu})$ discussed in Section 3.4.1, which corresponds to the solution of (3.27) using quadratic polynomials as basis functions, and a “naive” choice, v_{naive}^* , given as

$$v_{\text{naive}}^* \equiv \begin{cases} 1, & \text{on } \Gamma_{\text{east}}, \\ 0, & \text{at every other GLL node.} \end{cases} \quad (4.21)$$

Note that in a spectral element context, the naive choice would be the natural and computationally convenient choice to make for v^* .

In order to test the RB approximations, we introduce a test sample $\Xi_{\text{test}} \subset \mathcal{D}$ consisting of 200 random points with a uniform distribution over \mathcal{D} . We assume that Ξ_{test} is dense enough that the behavior of the RB output for $\boldsymbol{\mu} \in \Xi_{\text{test}}$ gives a good representation of the behavior of the RB output for all $\boldsymbol{\mu} \in \mathcal{D}$.

Evaluation of Flux Integral Outputs

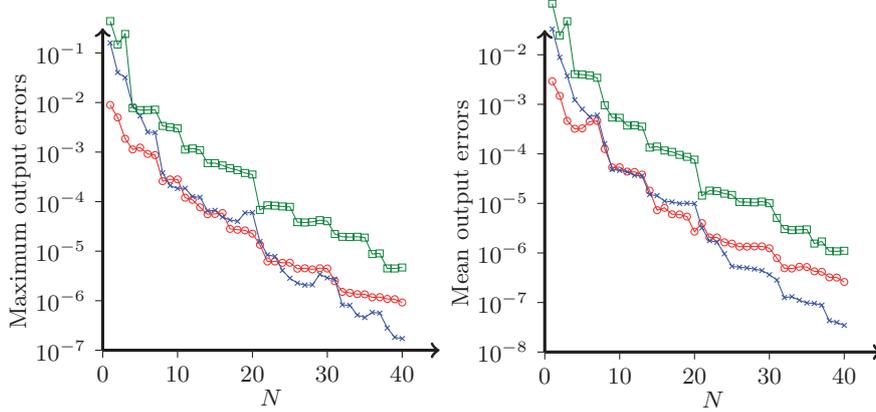


Figure 5: Maximum (left) and mean (right) of the output errors $e_N^{\text{out}}(\boldsymbol{\mu}; v^*)$ over Ξ_{test} for the three choices $\psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}})$ (\times), $\psi^{\mathcal{M}_2}(\boldsymbol{\mu})$ (\circ) and v_{naive}^* (\square) for v^* , as functions of the number of reduced basis functions, N .

In Figure 5, we plot the maximum (to the left) and mean (to the right) of the output errors $e_N^{\text{out}}(\boldsymbol{\mu}; \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}}))$, $e_N^{\text{out}}(\boldsymbol{\mu}; \psi^{\mathcal{M}_2}(\boldsymbol{\mu}))$ and $e_N^{\text{out}}(\boldsymbol{\mu}; v_{\text{naive}}^*)$ over all $\boldsymbol{\mu} \in \Xi_{\text{test}}$ as functions of the number of RB basis functions, N . We conclude that the two “good” choices for v^* in general perform about an order of magnitude better than the naive choice.

Next, for all $\boldsymbol{\mu} \in \Xi_{\text{test}}$, and for the particular case of $N = 25$, we compute the *effectivity* associated with the output error estimator $\Delta_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}$, defined as

$$\nu_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu}) \equiv \frac{\Delta_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu})}{e_N^{\text{out}}(\boldsymbol{\mu}; \psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}}))}. \quad (4.22)$$

Associated with the other two choices for v^* , we define the effectivities $\nu_{N, \mathcal{M}_2}^{\text{out}}(\boldsymbol{\mu})$ and $\nu_{N, v_{\text{naive}}^*}^{\text{out}}(\boldsymbol{\mu})$ in a similar way. For most $\boldsymbol{\mu} \in \Xi_{\text{test}}$, the effectivity $\nu_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu})$ is in the range $\mathcal{O}(100) < \nu_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu}) < \mathcal{O}(1000)$, as shown in Figure 6. For the other two effectivities, $\nu_{N, \mathcal{M}_2}^{\text{out}}(\boldsymbol{\mu})$ and $\nu_{N, v_{\text{naive}}^*}^{\text{out}}(\boldsymbol{\mu})$, the results are similar (not shown). We also find that, for most $\boldsymbol{\mu} \in \Xi_{\text{test}}$, $\nu_{N, \mathcal{M}_2}^{\text{out}}(\boldsymbol{\mu}) < \nu_{N, \bar{\boldsymbol{\mu}}}^{\text{out}}(\boldsymbol{\mu}) < \nu_{N, v_{\text{naive}}^*}^{\text{out}}(\boldsymbol{\mu})$. This is, however, not generally true for other choices of N .

The reason for the output error estimators being rather conservative is the large “angle” between the error of the primal problem, $u_N(\boldsymbol{\mu}) - u^{\mathcal{N}}(\boldsymbol{\mu})$, and the error of the dual problem, $\psi^{\mathcal{N}}(\bar{\boldsymbol{\mu}}) - \psi^{\mathcal{N}}(\boldsymbol{\mu})$. Thus, the Cauchy-Schwarz inequality, used in (3.23), becomes unsharp. This point is readily verified for the

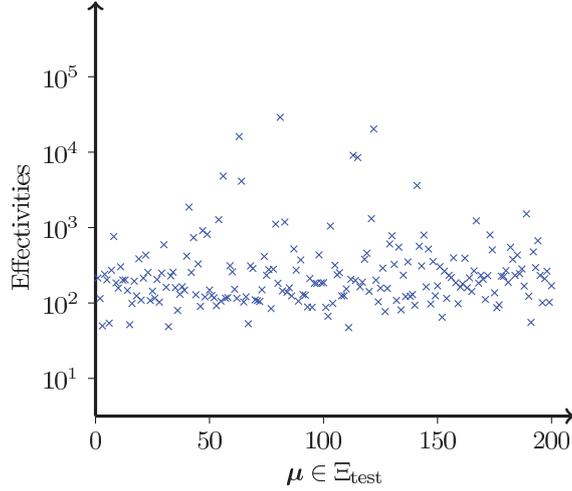


Figure 6: Output error bound effectivity $\nu_{N,\mu}^{\text{out}}(\mu)$ for all $\mu \in \Xi_{\text{test}}$ for $N = 25$ (in no particular order).

estimator $\Delta_{N,\bar{\mu}}^{\text{out}}$ by separate computation of the effectivities associated with the estimators $\Delta_N(\mu)$ and $\Delta_{\bar{\mu}}(\mu)$ for the primal and dual problems, respectively, which are indeed close to unity (of course, the same argument works for the other estimators as well). For $N = 25$, we find

$$\max_{\mu \in \Xi_{\text{test}}} \frac{\Delta_N(\mu)}{\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mu}} \approx 2.68 \quad (4.23)$$

and (irrespective of N)

$$\max_{\mu \in \Xi_{\text{test}}} \frac{\Delta_{\bar{\mu}}(\mu)}{\|\psi^{\mathcal{N}}(\mu) - \psi^{\mathcal{N}}(\bar{\mu})\|_{\mu}} \approx 2.94. \quad (4.24)$$

Hence, we do have a quite sharp bound for the right hand side of (3.23) for all $\mu \in \Xi_{\text{test}}$, and the unsharpness of the RB output error bound must be ascribed to the Cauchy-Schwarz inequality. Another implication of the sharpness of the individual error bounds is that our coercivity lower bound, $\alpha_{\text{LB}}(\mu)$, must be very sharp for all $\mu \in \Xi_{\text{test}}$.

4.4 A note on a special compliant problem

Since the governing equation for our problem is the Laplace equation (4.4), the parametrised bilinear form $a(\cdot, \cdot; \boldsymbol{\mu})$ is symmetric for each $\boldsymbol{\mu} \in \mathcal{D}$, and the only term that enters on the right-hand-side in the weak formulation is the Dirichlet lifting term $-a(u^{\text{D}}, v; \boldsymbol{\mu})$ (as in (4.16)). Now, in the very special case that we would like to evaluate the flux integral output over the same electrode on which a unity potential is imposed, we may choose $v^* = u^{\text{D}}$ as the flux lifting function. In this case, our RB output of interest is $s_N(\boldsymbol{\mu}; u^{\text{D}}) = a(u_N(\boldsymbol{\mu}), u^{\text{D}}; \boldsymbol{\mu})$. With $e_N(\boldsymbol{\mu}) = u^{\mathcal{N}}(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})$, we get

$$\begin{aligned} |s^{\mathcal{N}}(\boldsymbol{\mu}) - s_N(\boldsymbol{\mu}; u^{\text{D}})| &= |a(e_N(\boldsymbol{\mu}), u^{\text{D}}; \boldsymbol{\mu})| \\ &= |a(u^{\text{D}}, e_N(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(u^{\mathcal{N},0}(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}); \boldsymbol{\mu})| \\ &= |a(e_N(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}); \boldsymbol{\mu})| = \|e_N(\boldsymbol{\mu})\|_{\boldsymbol{\mu}}^2, \end{aligned} \quad (4.25)$$

where we use the symmetry of $a(\cdot, \cdot; \boldsymbol{\mu})$, then (4.13) (with $u^{\mathcal{N}}(\boldsymbol{\mu}) = u^{\mathcal{N},0}(\boldsymbol{\mu}) + u^{\text{D}}(\boldsymbol{\mu})$) and the fact that $e_N(\boldsymbol{\mu}) \in X^{\mathcal{N}}$, and finally again symmetry of $a(\cdot, \cdot; \boldsymbol{\mu})$ and Galerkin orthogonality. Hence, the RB output error converges quadratically with the energy-norm error without any simultaneous primal-dual treatment.

In the multi-electrode case, it is of little practical interest to evaluate the capacitance over the electrode with unity Dirichlet data, since this evaluation would only yield the *total* capacitance, as if we were to sum up the capacitances between the selected electrode and each of the other electrodes. However, for the sake of argument, suppose our system consists of only two electrodes. Then the exact output over one of the electrodes is equal to the exact output over the other, with a minus sign. We can thus choose to evaluate the output over the electrode with unity Dirichlet data (and multiply by (-1)).

We emphasise again that this compliant effect is restricted to the special case in which $f = 0$, the unity Dirichlet input electrode coincides with the output electrode and $a(\cdot, \cdot; \boldsymbol{\mu})$ is symmetric for each $\boldsymbol{\mu} \in \mathcal{D}$.

In [7], the numerical example discussed in Sections 4.1–4.3 is extended to incorporate three outputs (specifically, the capacitances between the south electrode, Γ_{south} , and each of the other electrodes, Γ_{east} , Γ_{north} and Γ_{west}). Also, several symmetries of the problem are exploited — which we have not done in this paper for the sake of simplicity of exposition — and the empirical interpolation method is used in order to achieve an efficient offline-online decoupling of the RB computations. Finally, we also mention [21], in which a very similar electrostatics problem is solved with the h - p finite element method. As in our

example, the outputs of interest are the capacitances corresponding to pairs of electrodes, and evaluation of the outputs both via a flux lifting function and by direct computation is considered.

5 Concluding Remarks

We have shown that the flux lifting function, which we call v^* , should be chosen with care when evaluating flux integral outputs from reduced basis approximations. Our two different proposals for a “good” v^* have been seen to give better results (a smaller RB output error) than a naive v^* in a simple (Laplace equation) numerical example. In contrast, we note that the naive v^* would have performed equally well as the “good” ones within a standard finite element context. (In fact, the naive choice is convenient in terms of implementation, and is thus often used in practice for the FE method.)

In the case of many (flux integral) outputs of interest that are all functionals of the same RB solution, a standard primal-dual error reduction technique may become too expensive. In this case, choosing a good v^* is important to make sure that the RB (primal only) output error is not unnecessarily large.

Acknowledgements

We would like to thank Professor A.T. Patera of MIT, and Dr D.B.P. Huynh for the many helpful comments and suggestions received throughout the work on this paper. The work has been supported by the Norwegian University of Science and Technology. The support is gratefully acknowledged.

Bibliography

- [1] I. Babuška and A. Miller. The post-processing approach in the finite element method—part 1: Calculation of displacements, stresses and other higher derivatives of the displacements. *International Journal for Numerical Methods in Engineering*, 20(6):1085–1109, 1984.
- [2] I. Babuška and M. Suri. The p and h - p versions of the finite element method, basic principles and properties. *SIAM Review*, 36(4):578–632, 1994.

- [3] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathématique*, 339(9):667–672, 2004.
- [4] C. Bernardi and Y. Maday. Polynomial approximation of some singular functions. *Applicable Analysis*, 42(1):1–32, 1991.
- [5] C. Bernardi and Y. Maday. Spectral methods. In *Handbook of numerical analysis*, volume V, pages 209–485. North-Holland, 1997.
- [6] G. F. Carey. Derivative calculation from finite element solutions. *Computer Methods in Applied Mechanics and Engineering*, 35(1):1–14, 1982.
- [7] J. L. Eftang. Reduced basis methods for partial differential equations. Master’s thesis, Norwegian University of Science and Technology, 2008.
- [8] W. J. Gordon and C. A. Hall. Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, 7(4):461–477, 1973.
- [9] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [10] P. M. Gresho, R. L. Lee, R. L. Sani, M. K. Maslanik, and B. E. Eaton. The consistent galerkin fem for computing derived boundary quantities in thermal and/or fluids problems. *International journal for numerical methods in fluids*, 7:371–394, 1987.
- [11] P. Grisvard. *Singularities in boundary value problems*, volume 22 of *Recherches en Mathématiques Appliquées [Research in Applied Mathematics]*. Masson, Paris, 1992.
- [12] W. Gui and I. Babuška. The h , p and h - p versions of the finite element method in 1 dimension. I. The error analysis of the p -version. *Numerische Mathematik*, 49(6):577–612, 1986.
- [13] A. E. Løvgrén, Y. Maday, and E. M. Rønquist. A reduced basis element method for the steady Stokes problem. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40(3):529–552, 2006.

Paper 1

- [14] Y. Maday, N. C. Nguyen, A. T. Patera, and G. S. H. Pau. A general multipurpose interpolation procedure: The magic points. *Communications in Pure and Applied Mathematics*, 8:383–404, 2009.
- [15] Y. Maday and A. T. Patera. Spectral element methods for the incompressible Navier-Stokes equations. In A. K. Noor and J. T. Oden, editors, *State of the art surveys in Computational Mechanics*, pages 71–143. American Society of Mechanical Engineers, 1989.
- [16] N. C. Nguyen. *Reduced-Basis Approximations and A Posteriori Error Bounds for Nonaffine and Nonlinear Partial Differential Equations: Application to Inverse Analysis*. PhD thesis, Singapore-MIT Alliance, National University of Singapore, 2005.
- [17] A. T. Patera and G. Rozza. *Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations, Version 1.0*. Copyright MIT 2006–2007, to appear in (tentative rubric) *MIT Pappalardo Graduate Monographs in Mechanical Engineering*. <http://augustine.mit.edu>.
- [18] C. Prud’homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici. Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *Journal of Fluids Engineering*, 124(1):70–80, 2002.
- [19] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*, volume 23 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1994.
- [20] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [21] B. A. Szabó. Solution of electrostatic problems by the p -version of the finite element method. *Communications in Applied Numerical Methods*, 4(4):565–572, 1988.
- [22] K. Veroy, C. Prud’homme, D. V. Rovas, and A. T. Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *Proceedings of the*

Evaluation of Flux Integral Outputs

*16th AIAA Computational Fluid Dynamics Conference, 2003. AIAA Paper
2003-3847.*

PAPER 2

**AN “*hp*” CERTIFIED REDUCED BASIS METHOD
FOR PARAMETRIZED ELLIPTIC PARTIAL
DIFFERENTIAL EQUATIONS**

JENS L. EFTANG, ANTHONY T. PATERA, AND EINAR M. RØNQUIST

Published in
SIAM Journal on Scientific Computing,
Vol. 32, No. 6 (2010), pp. 3170-3200

AN “ hp ” CERTIFIED REDUCED BASIS METHOD FOR PARAMETRIZED ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS

JENS L. EFTANG*, ANTHONY T. PATERA†, AND EINAR M. RØNQUIST*

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

†*Department of Mechanical Engineering,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA*

Abstract

We present a new “ hp ” parameter multidomain certified reduced basis (RB) method for rapid and reliable online evaluation of functional outputs associated with parametrized elliptic partial differential equations. We propose, and provide theoretical justification for, a new procedure for adaptive partition (“ h ”-refinement) of the parameter domain into smaller parameter subdomains: we pursue a hierarchical splitting of the parameter (sub)domains based on proximity to judiciously chosen parameter *anchor points* within each subdomain. Subsequently, we construct individual standard RB approximation spaces (“ p ”-refinement) over each subdomain. Greedy parameter sampling procedures and *a posteriori* error estimation play important roles in both the “ h ”-type and “ p ”-type stages of the new algorithm. We present illustrative numerical results for a convection-diffusion problem: the new “ hp ”-approach is considerably faster (respectively, more costly) than the standard “ p ”-type reduced basis method in the online (respectively, offline) stage.

1 Introduction

The certified reduced basis (RB) method provides a computational framework for rapid and reliable evaluation of outputs associated with parametrized partial differential equations. Given any *input* parameter vector—such as geometric factors or material property coefficients—the RB field approximation is constructed as a Galerkin linear combination of precomputed “truth” finite element

(FE) “snapshots” at optimally chosen parameter values; the RB output approximation is then evaluated as a functional of the RB field approximation. The methodology was originally introduced in [1, 21] and then further analyzed in [22, 23]; for a review of both earlier and more recent contributions, see [24].

For problems in which the field variable varies smoothly with the parameters, good RB approximations can be obtained with very few snapshots: the RB approximation converges exponentially fast [5, 8]. Furthermore, rigorous *a posteriori* upper bounds for the error in the RB approximation (with respect to the truth discretization) can be readily developed [24]. Finally, under an assumption on “affine” parameter dependence (perhaps only approximate [4, 13]), both the RB output approximation *and* the associated RB output error bound can be computed very efficiently by offline-online computational procedures [24]. The RB method is especially attractive in engineering contexts in which low marginal (online) computational cost is advantageous: “real-time”—such as parameter estimation [19] and optimal control—and “many-query”—such as multiscale [6] or stochastic simulation [7].

The RB approximation space is specifically constructed to provide accurate approximations for any parameter value in a predefined parameter domain. Hence, larger parameter domains typically induce larger RB spaces and greater computational cost. In this paper we propose, and provide theoretical justification for, a new procedure for adaptive partition (“*h*”-refinement) of the parameter domain into smaller parameter subdomains: we pursue a hierarchical splitting of the parameter (sub)domains based on proximity to judiciously chosen parameter *anchor points* within each subdomain. Subsequently, we construct individual standard RB approximation spaces (“*p*”-refinement) over each subdomain. Greedy sampling procedures and rigorous *a posteriori* error estimation play important roles in both the “*h*”-type and “*p*”-type stages of the algorithm.

In this new approach, the RB approximation associated with any new parameter value is, as always, constructed as a linear Galerkin combination of snapshots from the parameter (sub)domain in which this new parameter value resides. However, we expect the online computational cost of the new approach to be greatly reduced relative to the online cost of the standard RB approach due to the smaller parameter (sub)domains and hence lower-dimensional local RB approximation spaces associated with the “*hp*” approximation. The method should be particularly effective for problems in which the solution has very different structures in different regions of the parameter domain—problems for which a snapshot from one parameter region may be of limited value for the RB approximation in another parameter region.

The notion of parameter domain refinement within the model order reduction framework is considered in several earlier works. In [2, 3], a reduced-order parameter multielement “interpolation” procedure is introduced for aeroelasticity problems; this interpolation procedure and our approach here share a similar error-adaptive domain-decomposition foundation. However, the two approaches are quite different: interpolation on a manifold rather than Galerkin projection (here); parameter domain partition based on a Voronoi diagram rather than a hierarchical tree structure decomposition (here); heuristic error indicators rather than rigorous error bounds (here); and less strict rather than strict offline-online segregation (here). However, our own approach cannot yet treat problems of the complexity considered in [2, 3].

In other related work [14, 15, 25], adaptive *train sample* refinement is considered to render the Greedy parameter sampling procedure more efficient: richer samples are considered only as needed in the Greedy iterations [25] and only where needed in the parameter domain [14, 15]. Our approach invokes a similar technique: we include new points in the train sample within each subdomain at each new level of “*h*”-refinement; we thus effectively adapt the train sample to the more “difficult” parameter regions which require many subdomains. Reference [14] also proposes a multiple-bases (“*hp*”) approach which shares certain features with our approach here but also differs in several important ways in particular related to the “*h*”-refinement partition strategy.

In section 2 we give the general problem statement along with various definitions required throughout the paper. In section 3 we review the standard (“*p*”-type) RB method; in section 4 we present the new “*h*”-type RB method and provide an *a priori* convergence theory for a zeroth-order approximation in the one-parameter case; in section 5 we present the new “*hp*”-type RB method as a combination of the “*p*”- and “*h*”-type methods. In section 6 we present numerical results for a convection-diffusion model problem and in particular we compare the computational cost of the new “*hp*”-approach to the standard “*p*”-type method. We conclude in section 7 with some final remarks.

2 Problem statement

We shall consider linear, elliptic, coercive, second-order partial differential equations. We denote the physical domain by $\Omega \subset \mathbb{R}^2$, and we introduce the spaces $L^2(\Omega) = \{v : \int_{\Omega} v^2 d\Omega < \infty\}$, $H^1(\Omega) = \{v \in L^2(\Omega) : |\nabla v| \in L^2(\Omega)\}$, and $H_0^1(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$. We further define the space associated with the exact solution (hence e) $X^e \equiv X^e(\Omega)$ such that $H_0^1(\Omega) \subseteq X^e(\Omega) \subseteq H^1(\Omega)$.

Paper 2

We introduce a compact parameter domain $\mathcal{D} \subset \mathbb{R}^P$; a point in \mathcal{D} shall be denoted $\boldsymbol{\mu} = (\mu_1, \dots, \mu_P)$.

For each $\boldsymbol{\mu} \in \mathcal{D}$, $a(\cdot, \cdot; \boldsymbol{\mu})$ is an X^e -coercive and X^e -continuous bilinear form and $f(\cdot; \boldsymbol{\mu})$ is an X^e -bounded linear functional. To accommodate an efficient offline-online computational procedure, we assume that a and f admit affine expansions as

$$a(\cdot, \cdot; \boldsymbol{\mu}) = \sum_{q=1}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\boldsymbol{\mu}), \quad f(\cdot; \boldsymbol{\mu}) = \sum_{q=1}^{Q_f} f^q(\cdot) \Theta_f^q(\boldsymbol{\mu}) \quad (2.1)$$

for modest Q_a and Q_f , where the a^q and f^q are $\boldsymbol{\mu}$ -independent continuous bilinear forms and linear functionals, respectively, and the Θ_a^q and Θ_f^q are $\boldsymbol{\mu}$ -dependent continuous functions. (The assumption (2.1) can be relaxed with the *empirical interpolation* method [4, 13] for the construction of good affine approximations to a and f .) For simplicity, we introduce $Q = \max\{Q_a, Q_f\}$.

The exact problem statement reads as follows: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u^e(\boldsymbol{\mu}) \in X^e$ such that

$$a(u^e(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) \quad \forall v \in X^e. \quad (2.2)$$

The output of interest can then be evaluated as a functional of the field variable, say, $s(\boldsymbol{\mu}) = l(u^e(\boldsymbol{\mu}); \boldsymbol{\mu})$ for some X^e -bounded linear functional $l(\cdot; \boldsymbol{\mu})$. In this paper, however, for simplicity of exposition, we consider no particular output(s) of interest; our “hp” procedure does not depend on the output functional(s) chosen.

We next introduce a “truth” finite element (FE) space $X \equiv X^{\mathcal{N}}(\Omega) \subset X^e(\Omega)$ of finite dimension \mathcal{N} . The truth discretization of (2.2) reads as follows: For any $\boldsymbol{\mu} \in \mathcal{D}$, find $u(\boldsymbol{\mu}) \in X$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) \quad \forall v \in X. \quad (2.3)$$

We assume that X is rich enough that the error between the truth and exact solutions is in practice negligible. The RB approximation will be built upon truth snapshots $u(\boldsymbol{\mu}_n) \approx u^e(\boldsymbol{\mu}_n)$, $1 \leq n \leq N$, for judiciously chosen $\boldsymbol{\mu}_1 \in \mathcal{D}, \dots, \boldsymbol{\mu}_N \in \mathcal{D}$, and the RB error shall be measured with respect to the truth FE approximation.

For any $\boldsymbol{\mu} \in \mathcal{D}$, let $a_s(\cdot, \cdot; \boldsymbol{\mu})$ denote the symmetric part of $a(\cdot, \cdot; \boldsymbol{\mu})$: for any $\boldsymbol{\mu} \in \mathcal{D}$ and for all $v, w \in X$, $a_s(w, v; \boldsymbol{\mu}) = \frac{1}{2}(a(w, v; \boldsymbol{\mu}) + a(v, w; \boldsymbol{\mu}))$. Further, let $\bar{\boldsymbol{\mu}} \in \mathcal{D}$ denote a fixed *reference parameter*. We then define the parameter-independent X -inner product and corresponding X -norm as

$$(\cdot, \cdot)_X \equiv a_s(\cdot, \cdot; \bar{\boldsymbol{\mu}}), \quad \|\cdot\|_X = \sqrt{(\cdot, \cdot)_X}, \quad (2.4)$$

respectively. By our assumptions, $\|\cdot\|_X$ is equivalent to the H^1 -norm.

Finally, we introduce for all $\boldsymbol{\mu} \in \mathcal{D}$ the coercivity and continuity constants of $a(\cdot, \cdot; \boldsymbol{\mu})$ with respect to the X -norm,

$$\alpha(\boldsymbol{\mu}) \equiv \inf_{w \in X} \frac{a(w, w; \boldsymbol{\mu})}{\|w\|_X^2}, \quad \gamma(\boldsymbol{\mu}) \equiv \sup_{v \in X} \sup_{w \in X} \frac{a(v, w; \boldsymbol{\mu})}{\|v\|_X \|w\|_X}, \quad (2.5)$$

respectively. For any particular $\boldsymbol{\mu} \in \mathcal{D}$, we further require lower and upper bounds,

$$0 < \alpha_{\text{LB}}(\boldsymbol{\mu}) \leq \alpha(\boldsymbol{\mu}), \quad (2.6)$$

$$\infty > \gamma_{\text{UB}}(\boldsymbol{\mu}) \geq \gamma(\boldsymbol{\mu}), \quad (2.7)$$

which shall play a role in our computational procedures. We shall also invoke lower and upper bounds over \mathcal{D} ,

$$\underline{\alpha} = \min_{\boldsymbol{\mu} \in \mathcal{D}} \alpha(\boldsymbol{\mu}), \quad (2.8)$$

$$\bar{\gamma} = \max_{\boldsymbol{\mu} \in \mathcal{D}} \gamma(\boldsymbol{\mu}), \quad (2.9)$$

for the purposes of our theoretical arguments.

We shall later need the following lemma.

Lemma 1. *Let $\Theta_a^q : \mathcal{D} \rightarrow \mathbb{R}$, $1 \leq q \leq Q_a$, $\Theta_f^q : \mathcal{D} \rightarrow \mathbb{R}$, $1 \leq q \leq Q_f$, satisfy Lipschitz conditions*

$$|\Theta_a^q(\boldsymbol{\mu}_1) - \Theta_a^q(\boldsymbol{\mu}_2)| \leq C_a |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2| \quad \forall \boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathcal{D}, \quad 1 \leq q \leq Q_a, \quad (2.10)$$

$$|\Theta_f^q(\boldsymbol{\mu}_1) - \Theta_f^q(\boldsymbol{\mu}_2)| \leq C_f |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2| \quad \forall \boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathcal{D}, \quad 1 \leq q \leq Q_f. \quad (2.11)$$

Then, given any $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathcal{D}$, there exists a positive constant \tilde{C} such that

$$\|u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2)\|_X \leq \tilde{C} |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2|. \quad (2.12)$$

Proof. We have

$$a(u(\boldsymbol{\mu}_1), v; \boldsymbol{\mu}_1) = f(v; \boldsymbol{\mu}_1) \quad \forall v \in X, \quad (2.13)$$

$$a(u(\boldsymbol{\mu}_2), v; \boldsymbol{\mu}_2) = f(v; \boldsymbol{\mu}_2) \quad \forall v \in X. \quad (2.14)$$

By bilinearity of a , we thus have for all $v \in X$

$$\begin{aligned} a(u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2), v; \boldsymbol{\mu}_1) &= f(v; \boldsymbol{\mu}_1) - f(v; \boldsymbol{\mu}_2) \\ &\quad + a(u(\boldsymbol{\mu}_2), v; \boldsymbol{\mu}_2) - a(u(\boldsymbol{\mu}_2), v; \boldsymbol{\mu}_1). \end{aligned} \quad (2.15)$$

Paper 2

We first examine the right-hand side of (2.15).

By the triangle inequality and the affine expansions (2.1) for a and f , we have for all $w, v \in X$ and any $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathcal{D}$

$$|a(w, v; \boldsymbol{\mu}_1) - a(w, v; \boldsymbol{\mu}_2)| \leq \sum_{q=1}^{Q_a} |a^q(w, v)(\Theta_a^q(\boldsymbol{\mu}_1) - \Theta_a^q(\boldsymbol{\mu}_2))| \quad (2.16)$$

and

$$|f(v; \boldsymbol{\mu}_1) - f(v; \boldsymbol{\mu}_2)| \leq \sum_{q=1}^{Q_f} |f^q(v)(\Theta_f^q(\boldsymbol{\mu}_1) - \Theta_f^q(\boldsymbol{\mu}_2))|, \quad (2.17)$$

respectively. By our hypothesis, (2.10) and (2.11) on Θ_a^q , $1 \leq q \leq Q_a$, and Θ_f^q , $1 \leq q \leq Q_f$, respectively, and continuity of a^q , $1 \leq q \leq Q_a$, and f^q , $1 \leq q \leq Q_f$, there exist constants \tilde{c}_1 and \tilde{c}_2 (independent of $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$) such that

$$|a(w, v; \boldsymbol{\mu}_1) - a(w, v; \boldsymbol{\mu}_2)| \leq \tilde{c}_1 \|v\|_X \|w\|_X |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2| \quad (2.18)$$

and

$$|f(v; \boldsymbol{\mu}_1) - f(v; \boldsymbol{\mu}_2)| \leq \tilde{c}_2 \|v\|_X |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2|. \quad (2.19)$$

Recall that Q_a and Q_f are fixed and finite.

We now let $v = u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2)$ in (2.15) and deduce from the triangle inequality, (2.18), and (2.19) that

$$\begin{aligned} & a(u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2), u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2); \boldsymbol{\mu}_1) \\ & \leq (\tilde{c}_1 \|u(\boldsymbol{\mu}_2)\|_X + \tilde{c}_2) \|u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2)\|_X |\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2|. \end{aligned} \quad (2.20)$$

By coercivity and the bound (2.8), we get

$$\|u(\boldsymbol{\mu}_1) - u(\boldsymbol{\mu}_2)\|_X \leq \frac{1}{\underline{\alpha}} (\tilde{c}_1 \|u(\boldsymbol{\mu}_2)\|_X + \tilde{c}_2) |\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1|. \quad (2.21)$$

Finally, by the Lax–Milgram lemma, (2.8), the boundedness of $f(\cdot; \boldsymbol{\mu})$ for any $\boldsymbol{\mu} \in \mathcal{D}$, and the fact that $\mathcal{D} \subset \mathbb{R}^P$ is closed,

$$\|u(\boldsymbol{\mu}_2)\|_X \leq \frac{\|f(\cdot; \boldsymbol{\mu}_2)\|_{X'}}{\alpha_{\text{LB}}(\boldsymbol{\mu})} \leq \max_{\boldsymbol{\mu} \in \mathcal{D}} \frac{\|f(\cdot; \boldsymbol{\mu})\|_{X'}}{\underline{\alpha}} \quad (2.22)$$

(here X' denotes the dual space of X), and we thus obtain the desired result with

$$\tilde{C} = \max_{\boldsymbol{\mu} \in \mathcal{D}} \frac{\tilde{c}_1 \|f(\cdot; \boldsymbol{\mu})\|_{X'} + \tilde{c}_2 \underline{\alpha}}{\underline{\alpha}^2}. \quad (2.23)$$

(We can develop a constant \tilde{C} that is furthermore independent of \mathcal{N} by replacing the truth entities $\underline{\alpha}$ and $\|f(\cdot; \boldsymbol{\mu})\|_{X'}$ in (2.23) by the corresponding exact entities.) \square

3 The “ p ”-type RB method

In the standard RB approach, a single approximation space is enriched with new basis functions until the space is considered sufficiently rich; we shall refer to this approach as the “ p ”-type RB method. The new “ h ”-type and “ hp ”-type methods will borrow and adapt several of the ingredients from the standard approach: *a posteriori* error estimation; greedy parameter sampling; and offline-online computational decoupling of the RB discretization and the truth FE discretization through a construction–evaluation decomposition. Below, we summarize the standard RB method with particular emphasis on these key ingredients.

3.1 Approximation

The RB approximation space $X_N \equiv X_N(\Omega) \subset X^{\mathcal{N}}(\Omega)$ is defined in terms of a set of parameter vectors $\boldsymbol{\mu}_1 \in \mathcal{D}, \dots, \boldsymbol{\mu}_N \in \mathcal{D}$ as

$$X_N = \text{span}\{u(\boldsymbol{\mu}_1), \dots, u(\boldsymbol{\mu}_N)\}. \quad (3.1)$$

(Note that, in practice, an $(\cdot, \cdot)_X$ -orthonormal basis for X_N is constructed by a Gram–Schmidt procedure.) The RB approximation reads as follows: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u_N(\boldsymbol{\mu}) \in X_N$ such that

$$a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) \quad \forall v \in X_N. \quad (3.2)$$

Under the assumption that $u(\boldsymbol{\mu})$ depends smoothly on the parameters, we expect that N —the dimension of the RB space—can be chosen much smaller than \mathcal{N} —the dimension of the truth space X —for comparable numerical accuracy.

We finally *formally* define the “order” p of the RB approximation as $p \equiv N^{1/P} - 1$. We shall return to this point and elaborate on this interpretation in Remark 5.

3.2 *A posteriori* error estimation

We recall here the derivation of an *a posteriori* X -norm bound for the error in the RB field approximation relative to the corresponding truth approximation [24].

Given any $\boldsymbol{\mu} \in \mathcal{D}$, we obtain the RB approximation, $u_N(\boldsymbol{\mu})$, from (3.2); we then define for all $v \in X$ the RB residual as

$$r_N(v; \boldsymbol{\mu}) \equiv f(v; \boldsymbol{\mu}) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}). \quad (3.3)$$

The *Riesz representation* of the residual, $\mathcal{R}_N(\boldsymbol{\mu}) \in X$, satisfies

$$(\mathcal{R}_N(\boldsymbol{\mu}), v)_X = r_N(v; \boldsymbol{\mu}) \quad \forall v \in X. \quad (3.4)$$

We can now state the following lemma.

Lemma 2 (*a posteriori* X -norm error bound). *For any $\boldsymbol{\mu} \in \mathcal{D}$, the RB error bound*

$$\Delta_N(\boldsymbol{\mu}) \equiv \frac{\|\mathcal{R}_N(\boldsymbol{\mu})\|_X}{\alpha_{\text{LB}}(\boldsymbol{\mu})} \quad (3.5)$$

satisfies

$$\|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_X \leq \Delta_N(\boldsymbol{\mu}), \quad (3.6)$$

$$\frac{\Delta_N(\boldsymbol{\mu})}{\|u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})\|_X} \leq \frac{\gamma_{\text{UB}}(\boldsymbol{\mu})}{\alpha_{\text{LB}}(\boldsymbol{\mu})} \quad (3.7)$$

for $\alpha_{\text{LB}}(\boldsymbol{\mu})$ and $\gamma_{\text{UB}}(\boldsymbol{\mu})$ given by (2.6) and (2.7), respectively.

Proof. The RB error, $e_N(\boldsymbol{\mu}) = u(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu})$, satisfies the error-residual equation

$$a(e_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = r_N(v; \boldsymbol{\mu}) \quad \forall v \in X. \quad (3.8)$$

To obtain (3.6), we choose $e_N(\boldsymbol{\mu})$ for v in (3.8) and invoke (3.4) and the Cauchy–Schwarz inequality to get

$$a(e_N(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}); \boldsymbol{\mu}) = (\mathcal{R}_N(\boldsymbol{\mu}), e_N(\boldsymbol{\mu}))_X \leq \|\mathcal{R}_N(\boldsymbol{\mu})\|_X \|e_N(\boldsymbol{\mu})\|_X; \quad (3.9)$$

we then invoke coercivity and (2.6) to arrive at

$$\alpha_{\text{LB}}(\boldsymbol{\mu}) \|e_N(\boldsymbol{\mu})\|_X^2 \leq \|\mathcal{R}_N(\boldsymbol{\mu})\|_X \|e_N(\boldsymbol{\mu})\|_X. \quad (3.10)$$

The result (3.6) now directly follows from the definition (3.5).

To obtain (3.7), we choose $\mathcal{R}_N(\boldsymbol{\mu})$ for v in (3.8) and invoke (3.4), continuity, and (2.7) to get

$$\|\mathcal{R}_N(\boldsymbol{\mu})\|_X^2 = a(e_N(\boldsymbol{\mu}), \mathcal{R}_N(\boldsymbol{\mu}); \boldsymbol{\mu}) \leq \gamma_{\text{UB}}(\boldsymbol{\mu}) \|e_N(\boldsymbol{\mu})\|_X \|\mathcal{R}_N(\boldsymbol{\mu})\|_X; \quad (3.11)$$

hence $\|\mathcal{R}_N(\boldsymbol{\mu})\|_X / \|e_N(\boldsymbol{\mu})\|_X \leq \gamma_{\text{UB}}(\boldsymbol{\mu})$, and the result (3.7) follows from the definition (3.5). \square

3.3 Construction-evaluation decomposition

Thanks to the assumption (2.1) on affine parameter dependence, the computational procedures for the RB solution and error bound admit *construction-evaluation* decompositions (see also [18, 20]): the construction stage is computationally expensive—the operation count depends on \mathcal{N} ; however, in the subsequent evaluation stage we can then rapidly—independently of \mathcal{N} —evaluate the RB approximation and RB error bound for any $\boldsymbol{\mu} \in \mathcal{D}$. (In actual practice we would of course also evaluate the RB output and RB output error bound at negligible additional cost.) The construction-evaluation decomposition in turn permits the full offline-online computational decoupling described in the introduction; we further discuss this decoupling below.

We first describe the construction-evaluation decomposition for the RB approximation: Let $\{\zeta_1 \in X_N, \dots, \zeta_N \in X_N\}$ denote an X -orthonormal basis for X_N . In the construction stage, we assemble the matrices $A_N^q \in \mathbb{R}^{N \times N}$, $1 \leq q \leq Q_a$, and the vectors $F_N^q \in \mathbb{R}^N$, $1 \leq q \leq Q_f$, the elements of which are defined by

$$A_{N,mn}^q \equiv a^q(\zeta_n, \zeta_m), \quad F_{N,m}^q \equiv f^q(\zeta_m), \quad 1 \leq m, n \leq N, \quad (3.12)$$

respectively. In the evaluation stage, given any $\boldsymbol{\mu} \in \mathcal{D}$, we evaluate the functions $\Theta_a^q(\boldsymbol{\mu})$, $1 \leq q \leq Q_a$, and $\Theta_f^q(\boldsymbol{\mu})$, $1 \leq q \leq Q_f$, in $\mathcal{O}(Q)$ operations; we then construct the RB stiffness matrix and load vector as

$$A_N(\boldsymbol{\mu}) = \sum_{q=1}^{Q_a} \Theta_a^q(\boldsymbol{\mu}) A_N^q, \quad F_N(\boldsymbol{\mu}) = \sum_{q=1}^{Q_f} \Theta_f^q(\boldsymbol{\mu}) F_N^q, \quad (3.13)$$

respectively, in $\mathcal{O}(Q_a N^2 + Q_f N) = \mathcal{O}(QN^2)$ operations; finally, we solve the associated system of equations

$$A_N(\boldsymbol{\mu}) \underline{u}_N(\boldsymbol{\mu}) = F_N(\boldsymbol{\mu}) \quad (3.14)$$

for the RB basis coefficients $\underline{u}_N(\boldsymbol{\mu}) \equiv [u_{N,1}(\boldsymbol{\mu}), \dots, u_{N,N}(\boldsymbol{\mu})]^T$ in $\mathcal{O}(N^3)$ operations (we must anticipate that $A_N(\boldsymbol{\mu})$ is dense).

We next describe the construction-evaluation decomposition for the dual norm of the residual. By linearity, we can write (3.4) as

$$(\mathcal{R}_N(\boldsymbol{\mu}), v)_X = \sum_{q=1}^{Q_f} \Theta_f^q(\boldsymbol{\mu}) f^q(v) - \sum_{q=1}^{Q_a} \sum_{n=1}^N \Theta_a^q(\boldsymbol{\mu}) u_{N,n}(\boldsymbol{\mu}) a^q(\zeta_n, v) \quad (3.15)$$

$$\equiv \sum_{n=1}^{\hat{N}} \Gamma_n(\boldsymbol{\mu}) \mathcal{L}_n(v), \quad (3.16)$$

where $\hat{N} = Q_f + NQ_a$. By linear superposition, we can thus write

$$\mathcal{R}_N(\boldsymbol{\mu}) = \sum_{n=1}^{\hat{N}} \Gamma_n(\boldsymbol{\mu}) \mathcal{G}_n, \quad (3.17)$$

where, for $1 \leq n \leq \hat{N}$,

$$(\mathcal{G}_n, v)_X = \mathcal{L}_n(v) \quad \forall v \in X. \quad (3.18)$$

We thus have

$$\|\mathcal{R}_N(\boldsymbol{\mu})\|_X^2 = (\mathcal{R}_N(\boldsymbol{\mu}), \mathcal{R}_N(\boldsymbol{\mu}))_X \quad (3.19)$$

$$= \sum_{m=1}^{\hat{N}} \sum_{n=1}^{\hat{N}} \Gamma_m(\boldsymbol{\mu}) \Gamma_n(\boldsymbol{\mu}) G_{mn}, \quad (3.20)$$

where the G_{mn} are defined as

$$G_{mn} \equiv (\mathcal{G}_m, \mathcal{G}_n)_X, \quad 1 \leq m, n \leq \hat{N}. \quad (3.21)$$

In the construction stage we first perform the truth Poisson solves (3.18) for \mathcal{G}_n , $1 \leq n \leq \hat{N}$; we then compute and store the inner products G_{mn} , $1 \leq m, n \leq \hat{N}$. In the evaluation stage, we evaluate the functions $\Gamma_n(\boldsymbol{\mu})$, $1 \leq n \leq \hat{N}$, in $\mathcal{O}(NQ_a + Q_f) = \mathcal{O}(NQ)$ operations and then perform the summation (3.19) in $\mathcal{O}((NQ_a + Q_f)^2) = \mathcal{O}(N^2Q^2)$ operations.

In general, the coercivity lower bound $\alpha_{\text{LB}}(\boldsymbol{\mu})$ will not be known analytically and must be computed. An efficient construction-evaluation decomposition for the coercivity lower bound—the *successive constraint method*—can be found in [17, 24]; the evaluation complexity is independent of \mathcal{N} . We do not discuss this component further here in particular because for our numerical example of section 6 an analytical lower bound $\alpha_{\text{LB}}(\boldsymbol{\mu})$ is, in fact, available.

Algorithm 1 Greedy¹($\Xi, \boldsymbol{\mu}_1, \epsilon_{\text{tol}}, \tilde{N}_{\text{max}}$).

initialize: $N \leftarrow 0, \epsilon_0 \leftarrow \infty, X_0 \leftarrow \text{span}\{0\}$
while $\epsilon_N > \epsilon_{\text{tol}}$ and $N < \tilde{N}_{\text{max}}$ **do**
 $N \leftarrow N + 1$
 $X_N \leftarrow X_{N-1} \oplus \text{span}\{u(\boldsymbol{\mu}_N)\}$
 $\epsilon_N \leftarrow \max_{\boldsymbol{\mu} \in \Xi} \Delta_N(\boldsymbol{\mu})$
 $\boldsymbol{\mu}_{N+1} \leftarrow \arg \max_{\boldsymbol{\mu} \in \Xi} \Delta_N(\boldsymbol{\mu})$
end while
 $N_{\text{max}} \leftarrow N$

3.4 Greedy parameter sampling

We now discuss the construction of the hierarchical RB approximation spaces $X_N = \text{span}\{u(\boldsymbol{\mu}_n)\}_{n=1}^N \subset X$, $1 \leq N \leq N_{\text{max}}$ (see also [24, 26]). We first introduce a finite *train sample* $\Xi \subset \mathcal{D}$; a (random, say) initial parameter vector $\boldsymbol{\mu}_1 \in \mathcal{D}$; an error tolerance ϵ_{tol} ; and a maximum RB dimension \tilde{N}_{max} . We then perform Algorithm 1. The output of the algorithm is the RB space $X_{N_{\text{max}}}$, where $N_{\text{max}} \leq \tilde{N}_{\text{max}}$. Note that the construction-evaluation decomposition allows us to use a dense train sample: each evaluation of the error bound in the max is very inexpensive; the truth is invoked only for the “winning” candidates, $\boldsymbol{\mu}_N$, $1 \leq N \leq N_{\text{max}}$.

3.5 Offline-online computational decoupling

We now describe the full offline-online decoupling procedure for the “ p ”-type RB approximation: the offline stage—performed only once as preprocessing—may be very expensive (\mathcal{N} -dependent); however, the subsequent (\mathcal{N} -independent) online stage—performed many times for the computation of the RB solution (and output) and RB error bound (and output error bound)—is very fast.

The offline stage is essentially the Greedy¹ algorithm (Algorithm 1). The parameter-independent entities $A_{N_{\text{max}}}^q \in \mathbb{R}^{N_{\text{max}} \times N_{\text{max}}}$, $1 \leq q \leq Q_a$, $F_{N_{\text{max}}}^q \in \mathbb{R}^{N_{\text{max}}}$, $1 \leq q \leq Q_f$, and $(\mathcal{G}_m, \mathcal{G}_n)_X$, $1 \leq m, n \leq Q_a N_{\text{max}} + Q_f$, are retained from the construction stage of the last iteration. The permanent *online* storage requirement is thus $\mathcal{O}(Q_a N_{\text{max}}^2 + Q_f N_{\text{max}}) = \mathcal{O}(Q N_{\text{max}}^2)$ for the $A_{N_{\text{max}}}^q$ and $F_{N_{\text{max}}}^q$, and $\mathcal{O}((Q_a N_{\text{max}} + Q_f)^2) = \mathcal{O}(Q^2 N_{\text{max}}^2)$ for the $(\mathcal{G}_m, \mathcal{G}_n)_X$. We note that since the RB spaces are nested, we can extract subarrays from the stored entities in order to construct RB approximations of any order $1 \leq N \leq N_{\text{max}}$ (hence providing for online adaptivity).

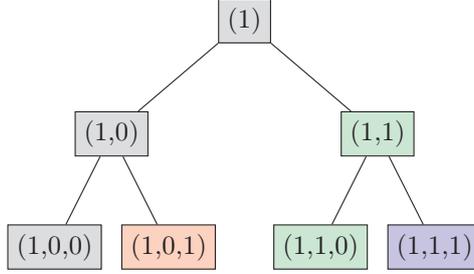


Figure 1: A perfect binary tree and associated Boolean vectors corresponding to $L = 3$.

The online stage is, for the “ p ”-type method, equivalent to the evaluation stage: given any $\boldsymbol{\mu} \in \mathcal{D}$, we assemble the RB system in $\mathcal{O}(Q_a N^2 + Q_f N) = \mathcal{O}(QN^2)$ operations, compute the RB solution in $\mathcal{O}(N^3)$ operations, and finally evaluate the RB error bound in $\mathcal{O}((NQ_a + Q_f)^2) = \mathcal{O}(N^2 Q^2)$ operations.

4 The “ h ”-type RB method

In this section we formulate the “ h ”-type RB method. We first provide preliminaries required throughout the paper; we next present the “ h ”-type approximation algorithm; we then consider *a posteriori* error estimation; we subsequently describe the offline-online computational decomposition; finally, we develop a new *a priori* convergence theory for the zeroth-order approximation in the case of one parameter.

4.1 Preliminaries: Tree-subdomain structure

We first define the set of Boolean vectors of length l ,

$$\mathcal{B}_l \equiv \{1\} \times \{0, 1\}^{l-1}; \quad (4.1)$$

we shall denote a particular member of \mathcal{B}_l as

$$B_l = (1, i_2, \dots, i_l) \in \mathcal{B}_l. \quad (4.2)$$

We next introduce a perfect binary tree with L levels and $K = 2^{L-1}$ leaf nodes as shown in Figure 1 (for the particular case $L = 3$); we then associate each

$B_l \in \mathcal{B}_l$, $1 \leq l \leq L$, to a node in the tree. We note that appending a “0” to a vector B_l corresponds to a left bend in the tree and appending a “1” to a vector B_l corresponds to a right bend in the tree; we define these “bends” by the concatenation

$$(B_l, i) \equiv (1, i_2, \dots, i_l, i), \quad i \in \{0, 1\}, \quad (4.3)$$

and we say that B_l is the parent of the two children (B_l, i) , $i \in \{0, 1\}$.

Given an initial parameter domain \mathcal{D} , we shall perform the “ h ”-refinement by recursive splitting of \mathcal{D} into smaller parameter subdomains. The subdomains will be defined hierarchically and thus can be associated to a tree; we assume for the moment that we can organize the $2^L - 1$ subdomains in a *perfect* binary tree. We denote the subdomains as

$$\mathcal{V}_{B_l} \subset \mathcal{D}, \quad B_l \in \mathcal{B}_l, \quad 1 \leq l \leq L, \quad (4.4)$$

and we require the parent-child hierarchy

$$\mathcal{V}_{(B_l, 0)} \subset \mathcal{V}_{B_l}, \quad (4.5)$$

$$\mathcal{V}_{(B_l, 1)} \subset \mathcal{V}_{B_l}. \quad (4.6)$$

We associate to each subdomain \mathcal{V}_{B_l} a set of \bar{N} parameter values denoted by

$$\mathcal{M}_{\bar{N}, B_l} = \{\boldsymbol{\mu}_{1, B_l}, \dots, \boldsymbol{\mu}_{\bar{N}, B_l}\}, \quad B_l \in \mathcal{B}_l, \quad 1 \leq l \leq L, \quad (4.7)$$

in which $\boldsymbol{\mu}_{1, B_l} \in \mathcal{V}_{B_l}, \dots, \boldsymbol{\mu}_{\bar{N}, B_l} \in \mathcal{V}_{B_l}$; we may then define the RB approximation spaces (of dimension \bar{N}) associated with the “models” $\mathcal{M}_{\bar{N}, B_l}$ and subdomains \mathcal{V}_{B_l} , $B_l \in \mathcal{B}_l$, $1 \leq l \leq L$, as

$$X_{\bar{N}, B_l} = \text{span}\{u(\boldsymbol{\mu}_{1, B_l}), \dots, u(\boldsymbol{\mu}_{\bar{N}, B_l})\}, \quad B_l \in \mathcal{B}_l, \quad 1 \leq l \leq L. \quad (4.8)$$

(The actual bases are, as always, orthonormalized.)

To each model $\mathcal{M}_{\bar{N}, B_l}$ and corresponding subdomain \mathcal{V}_{B_l} we associate a parameter *anchor point*, $\hat{\boldsymbol{\mu}}_{B_l}$, defined as

$$\hat{\boldsymbol{\mu}}_{B_l} \equiv \boldsymbol{\mu}_{1, B_l}. \quad (4.9)$$

We shall further impose (by construction) that, for $1 \leq l \leq L - 1$,

$$\hat{\boldsymbol{\mu}}_{(B_l, 0)} = \hat{\boldsymbol{\mu}}_{B_l}, \quad (4.10)$$

$$\hat{\boldsymbol{\mu}}_{(B_l, 1)} \neq \hat{\boldsymbol{\mu}}_{B_l}; \quad (4.11)$$

Paper 2

the anchor point is thus inherited only by the “left” child. The partition of \mathcal{D} into subdomains is inferred from proximity to the anchor points.

To this end, we introduce for any Boolean vector $B_l \in \mathcal{B}_l$, $1 \leq l \leq L$, a “proximity function” $d_{B_l} : \mathcal{D} \rightarrow \mathbb{R}^+$. For example, we can choose the Euclidean distance between two points,

$$d_{B_l}(\boldsymbol{\mu}) = \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}_{B_l}\|_2. \quad (4.12)$$

We then successively evaluate the proximity function to determine, for any given $1 \leq l \leq L$, which subdomain $\mathcal{V}_{(1, i_2^*, \dots, i_l^*)} \subset \mathcal{D}$ contains a given $\boldsymbol{\mu} \in \mathcal{D}$,

$$\begin{aligned} i_2^* &= \arg \min_{i \in \{0,1\}} d_{(1,i)}(\boldsymbol{\mu}), \\ i_3^* &= \arg \min_{i \in \{0,1\}} d_{(1, i_2^*, i)}(\boldsymbol{\mu}), \\ &\vdots \\ i_l^* &= \arg \min_{i \in \{0,1\}} d_{(1, i_2^*, \dots, i_{l-1}^*, i)}(\boldsymbol{\mu}). \end{aligned} \quad (4.13)$$

We discuss the computational complexity shortly.

In general, the partition will not have the same number of refinement levels along every branch of the associated binary tree: in practice, the tree is not necessarily perfect. In this case, L shall denote the maximum number of levels in the tree—the *tree depth*. Branches are terminated with “empty leaf models” $\mathcal{M}_{\bar{N}, (B_l, 0)} = \mathcal{M}_{\bar{N}, (B_l, 1)} = \emptyset$ for some $B_l \in \mathcal{B}_l$, $1 \leq l \leq L$. (For any such B_l associated with empty child models we adopt the convention $d_{(B_l, 0)} = d_{(B_l, 1)} \equiv \infty$; we then terminate the search (4.13) whenever $d_{(1, i_2^*, \dots, i)}(\boldsymbol{\mu}) = \infty$ (for $i = 0, 1$.) We now define K as the number of leaf nodes in the tree (*exclusive* of the terminator empty models). The uniformity of the tree associated with the partition of \mathcal{D} can be measured by a *relative tree depth*

$$\eta_{\text{depth}} = \frac{\text{tree depth } (= L)}{\log_2 K + 1}; \quad (4.14)$$

note that $\eta_{\text{depth}} \geq 1$ and that $\eta_{\text{depth}} = 1$ corresponds to a perfect binary tree.

In what follows we shall need Algorithm 2, which is largely a restatement of Greedy¹ (Algorithm 1) restricted to a particular subdomain \mathcal{V}_{B_l} for given $B_l \in \mathcal{B}_l$. The evaluation of the *a posteriori* error bound $\Delta_N^{h\text{RB}}$ (to be defined shortly) is now performed over $\Xi_{B_l} \subset \mathcal{V}_{B_l}$. The output of the algorithm is an RB space X_{N_{\max}, B_l} and an associated model $\mathcal{M}_{N_{\max}, B_l, B_l}$. Note that even for

Algorithm 2 Greedy²($\Xi_{B_l}, \boldsymbol{\mu}_{1,B_l}, \epsilon_{\text{tol}}, \tilde{N}_{\max,B_l}$).

initialize: $N \leftarrow 0, \epsilon_{0,B_l} \leftarrow \infty, X_{0,B_l} \leftarrow \text{span}\{0\}, \mathcal{M}_{0,B_l} \leftarrow \emptyset$

while $\epsilon_N > \epsilon_{\text{tol}}$ and $N < \tilde{N}_{\max,B_l}$ **do**

$N \leftarrow N + 1$

$X_{N,B_l} \leftarrow X_{N-1,B_l} \oplus \text{span}\{u(\boldsymbol{\mu}_{N,B_l})\}$

$\mathcal{M}_{N,B_l} \leftarrow \mathcal{M}_{N-1,B_l} \cup \{\boldsymbol{\mu}_{N,B_l}\}$

$\epsilon_{N,B_l} \leftarrow \max_{\boldsymbol{\mu} \in \Xi_{B_l}} \Delta_N^{\text{hRB}}(\boldsymbol{\mu})$

$\boldsymbol{\mu}_{N+1,B_l} \leftarrow \arg \max_{\boldsymbol{\mu} \in \Xi_{B_l}} \Delta_N^{\text{hRB}}(\boldsymbol{\mu})$

end while

$N_{\max,B_l} \leftarrow N$

$\tilde{N}_{\max,B_l} = 1$ we perform one pass of the whole loop and hence identify (and retain) $\boldsymbol{\mu}_{2,B_l}$; however, in general, we compute only at most \tilde{N}_{\max,B_l} snapshots. For the “*h*”-type RB approximation of this section we shall require $N_{\max,B_l} \equiv \tilde{N}_{\max,B_l} \equiv \bar{N}$ for all B_l .

4.2 Approximation

We now introduce the equi-order “*h*”-type RB approximation algorithm. We start from the original parameter domain $\mathcal{V}_{(1)} = \mathcal{D}$ ($l = 1, B_l = (1)$); we introduce a finite train sample $\Xi_{(1)} \subset \mathcal{V}_{(1)}$; we choose an initial parameter anchor point $\hat{\boldsymbol{\mu}}_{(1)} \in \mathcal{D}$; we choose the error tolerance ϵ_{tol}^1 ; we set the desired maximum RB space dimension $\bar{N} \geq 1$. The partition is then determined as follows.

1. Construct a model with \bar{N} parameter values for the current subdomain \mathcal{V}_{B_l} with the standard Greedy² algorithm (Algorithm 2). The RB space $X_{\bar{N},B_l}$ and the model $\mathcal{M}_{\bar{N},B_l}$ are outputs from Greedy²($\Xi_{B_l}, \hat{\boldsymbol{\mu}}_{B_l}, 0, \bar{N}$). Note that we set the argument $\epsilon_{\text{tol}} = 0$ to enforce an RB space of dimension \bar{N} .¹
2. Compute the maximum *a posteriori* error bound over the train sample

¹We assume here that the parametric manifold $\mathcal{M} = \{u(\boldsymbol{\mu}), \boldsymbol{\mu} \in \mathcal{D}\}$ cannot be approximated exactly by $\bar{N} \ll \mathcal{N}$ snapshots. We also assume here that the train sample over each subdomain is sufficiently rich.

Paper 2

associated with the current subdomain

$$\epsilon_{\bar{N}, B_l} = \max_{\boldsymbol{\mu} \in \Xi_{B_l}} \Delta_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu}). \quad (4.15)$$

3. If $\epsilon_{\bar{N}, B_l} < \epsilon_{\text{tol}}^1$, the refinement is sufficiently good. Set

$$\mathcal{M}_{\bar{N}, (1, i_2, \dots, i_l, 0)} = \emptyset, \quad (4.16)$$

$$\mathcal{M}_{\bar{N}, (1, i_2, \dots, i_l, 1)} = \emptyset; \quad (4.17)$$

we thus terminate the branch of the associated binary tree.

4. If $\epsilon_{\bar{N}, B_l} \geq \epsilon_{\text{tol}}^1$, do the following.

(i) Define anchor points for two new models $\mathcal{M}_{\bar{N}, (B_l, 0)}$ and $\mathcal{M}_{\bar{N}, (B_l, 1)}$ as $\hat{\boldsymbol{\mu}}_{(B_l, 0)} = \hat{\boldsymbol{\mu}}_{B_l}$ and $\hat{\boldsymbol{\mu}}_{(B_l, 1)} = \boldsymbol{\mu}_{2, B_l}$, respectively. The model $\mathcal{M}_{\bar{N}, (B_l, 0)}$ inherits the anchor point from its “parent,” while the model $\mathcal{M}_{\bar{N}, (B_l, 1)}$ takes as anchor point the first parameter value chosen by the Greedy² algorithm; in the sense of the *a posteriori* error estimator, these two points are *maximally different* and hence good places to “anchor” the new models. When $\bar{N} \geq 2$ the remaining $\bar{N} - 2$ parameter values of $\mathcal{M}_{\bar{N}, B_l}$, as well as the associated snapshots and approximation spaces, are discarded.

(ii) Define a new and denser train sample $\tilde{\Xi}_{B_l} \subset \mathcal{V}_{B_l}$ of size $|\tilde{\Xi}_{B_l}| = 2|\Xi_{(1)}|$. (The temporary sample $\tilde{\Xi}_{B_l}$ is thus twice as large as the initial train sample.)

(iii) Construct $\Xi_{(B_l, 0)} \subset \mathcal{V}_{(B_l, 0)}$ and $\Xi_{(B_l, 1)} \subset \mathcal{V}_{(B_l, 1)}$ from $\tilde{\Xi}_{B_l}$ based on proximity to $\hat{\boldsymbol{\mu}}_{(B_l, 0)}$ and $\hat{\boldsymbol{\mu}}_{(B_l, 1)}$, respectively: a point $\boldsymbol{\mu} \in \tilde{\Xi}_{B_l}$ belongs to $\Xi_{(B_l, 0)}$ if and only if $d_{(B_l, 0)}(\boldsymbol{\mu}) \leq d_{(B_l, 1)}(\boldsymbol{\mu})$; otherwise $\boldsymbol{\mu}$ belongs to $\Xi_{(B_l, 1)}$.

5. Split the current branch into two new branches: set $B_{l+1}^{\text{left}} = (B_l, 0)$ and $B_{l+1}^{\text{right}} = (B_l, 1)$; update $l \leftarrow l + 1$ and proceed to step 1 first for $B_l = B_l^{\text{left}}$ and then for $B_l = B_l^{\text{right}}$.

The procedure may be more precisely defined by $h\text{RB}(\Xi_{(1)}, \hat{\boldsymbol{\mu}}_{(1)}, \bar{N}, \epsilon_{\text{tol}}^1)$, where $h\text{RB}$ is the recursive function defined in Algorithm 3. The output from Algorithm 3 is K subdomains associated with the K leaf nodes of the binary tree (exclusive of the terminator empty models). Each subdomain is associated to an

Algorithm 3 $h\text{RB}(\Xi_{B_l}, \hat{\boldsymbol{\mu}}_{B_l}, \bar{N}, \epsilon_{\text{tol}}^1)$.

Find $X_{\bar{N}, B_l}$ and $\mathcal{M}_{\bar{N}, B_l}$ from $\text{Greedy}^2(\Xi_{B_l}, \hat{\boldsymbol{\mu}}_{B_l}, \infty, \bar{N})$
 $\epsilon_{\bar{N}, B_l} \leftarrow \max_{\boldsymbol{\mu} \in \Xi_{B_l}} \Delta_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu})$;
if $\epsilon_{\bar{N}, B_l} < \epsilon_{\text{tol}}^1$ **then**
 Terminate branch: $\mathcal{M}_{\bar{N}, (B_l, 0)} = \emptyset$ and $\mathcal{M}_{\bar{N}, (B_l, 1)} = \emptyset$
else
 Define $\hat{\boldsymbol{\mu}}_{(B_l, 0)} = \boldsymbol{\mu}_{1, B_l}$ and $\hat{\boldsymbol{\mu}}_{(B_l, 1)} = \boldsymbol{\mu}_{2, B_l}$
 Construct $\Xi_{(B_l, 0)} \subset \mathcal{V}_{(B_l, 0)}$ and $\Xi_{(B_l, 1)} \subset \mathcal{V}_{(B_l, 1)}$
 $h\text{RB}(\Xi_{(B_l, 0)}, \hat{\boldsymbol{\mu}}_{(B_l, 0)}, \bar{N}, \epsilon_{\text{tol}}^1)$
 $h\text{RB}(\Xi_{(B_l, 1)}, \hat{\boldsymbol{\mu}}_{(B_l, 1)}, \bar{N}, \epsilon_{\text{tol}}^1)$
end if

\bar{N} -parameter model and an \bar{N} -dimensional approximation space. We emphasize that the *intermediate* models and approximation spaces—associated with *non-leaf* nodes at earlier levels in the tree—are discarded and do not “survive” with respect to the online stage. Finally, we note that the depth of the tree, L , is simply the number of nodes in the longest branch (exclusive of the terminator empty models).

Remark 1 (train sample refinement). *In step 4(ii) in the algorithm above, additional points are added to the train sample such that the number of points in the two new train samples will be roughly the same as in the old train sample, and in particular always much larger than \bar{N} . As a result, the “global” train sample over \mathcal{D} —the union of all the points in the train samples over all parameter subdomains—is adaptively refined as the “h”-type RB approximation becomes more accurate: the train sample is denser in regions of \mathcal{D} with smaller subdomains. We thus effectively include more train points where the solution varies more rapidly with the parameters.*

The train sample refinement is performed by a simple accept-reject Monte Carlo procedure: we draw from the uniform distribution over \mathcal{D} ; we then use the search (4.13) to determine whether a point belongs to a subdomain and thus can be included as a new point in the associated train sample. In the case that the proximity function is Euclidean distance (as in (4.12)), we need not sample from the entire parameter domain \mathcal{D} : we first compute the bounding box of the old train sample; we then sample the new points from a larger box that contains the bounding box with some safety margin—the assumption is that this larger box contains the entire subdomain. In the case in which the proximity function is the error bound (as we describe shortly), we sample from the entire domain

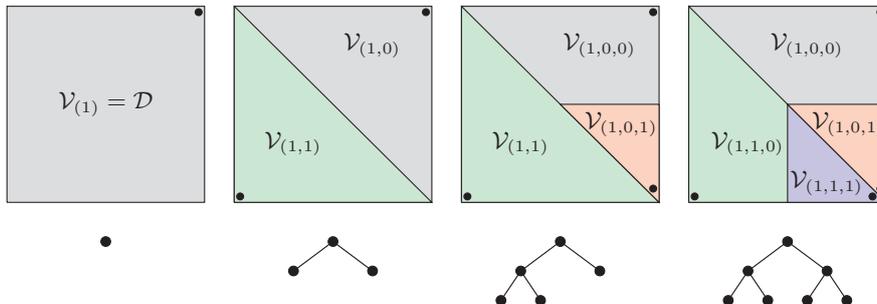


Figure 2: Two levels of “ h ”-refinement and associated binary tree; here $L = 3$.

\mathcal{D} since we have no a priori knowledge of the shape or connectedness of the subdomains.

An alternative and clearer approach to the train sample refinement 4(ii)–4(iii) might be to first split the current train sample into train samples associated with each subdomain and then enrich each of these samples to achieve size $|\Xi_{(1)}|$. However, for the numerical results in this paper we pursue the “first enrich then split” procedure described in step 4 above. We note that as long as the train samples are sufficiently rich the particular refinement procedure will not affect the numerical results significantly.

Remark 2 (“redundant” truth solves). *The Greedy algorithm—in particular in the case of a low-order (small \bar{N}) approximation—is likely to choose parameter values close to the boundaries of the parameter subdomains. As a result, two or more models may comprise some identical (or nearly identical) parameter values, and thus some of the offline truth solves are in some sense redundant. One way to reduce this snapshot redundancy is to share basis functions between approximation spaces if the associated greedily selected parameter values are sufficiently close. The development of an efficient algorithm for automatic sharing of basis functions is the subject of future work.*

In Figure 2 we illustrate the first two levels of “ h ”-refinement together with the associated binary tree for an “ h ”-type approximation with $\bar{N} = 1$. The initial domain is $\mathcal{V}_{(1)} = \mathcal{D}$ and the initial model is $\mathcal{M}_{1,(1)} = \{\hat{\boldsymbol{\mu}}_{(1)}\}$, where the anchor point $\hat{\boldsymbol{\mu}}_{(1)} = \boldsymbol{\mu}_{1,(1)}$ is chosen as the upper-right corner of the parameter domain. The method then greedily chooses the point $\boldsymbol{\mu}_{2,(1)}$ near the lower-left corner of $\mathcal{V}_{(1)}$; the initial anchor point is then relabeled as $\hat{\boldsymbol{\mu}}_{(1,0)} = \hat{\boldsymbol{\mu}}_{(1)}$ and the greedily chosen point is relabeled as $\hat{\boldsymbol{\mu}}_{(1,1)} = \boldsymbol{\mu}_{2,(1)}$. We now define

two new models $\mathcal{M}_{1,(1,0)} = \{\hat{\boldsymbol{\mu}}_{(1,0)}\}$ and $\mathcal{M}_{1,(1,1)} = \{\hat{\boldsymbol{\mu}}_{(1,1)}\}$, whose associated subdomains $\mathcal{V}_{(1,0)}$ and $\mathcal{V}_{(1,1)}$ are determined from proximity—here Euclidean distance—to the two anchor points. Next, $\mathcal{V}_{(1,1)}$ and $\mathcal{V}_{(1,0)}$ are partitioned in the same fashion (we assume here that the tolerance is satisfied within $\mathcal{V}_{(1,0,0)}$ and $\mathcal{V}_{(1,0,1)}$).

Finally, we may now define the “ h ”-type RB approximation. Given any $\boldsymbol{\mu} \in \mathcal{D}$ we first determine the subdomain $\mathcal{V}_{B_l^*}$ containing $\boldsymbol{\mu}$ from the search (4.13); for a perfect binary tree, $l = L$; however, more generally $l \leq L$. We then find $u_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu}) \in X_{\bar{N}, B_l^*}$ such that

$$a(u_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) \quad \forall v \in X_{\bar{N}, B_l^*}. \quad (4.18)$$

(Note that B_l^* depends on $\boldsymbol{\mu}$.) We discuss computational complexity shortly. Finally, we *formally* define the “order” of the “ h ”-type approximation as $p \equiv \bar{N}^{1/P} - 1$. We elaborate on this interpretation in Remark 5.

4.3 *A posteriori* error estimation

We can apply the same *a posteriori* bound developed for the “ p ”-type RB approximation in section 3.2 to the “ h ”-type (and below, “ hp ”-type) RB approximation. However, we shall require some new notation for the “ h ”-type error bound.

Given any $\boldsymbol{\mu} \in \mathcal{D}$ and a partition of \mathcal{D} into subdomains, we determine B_l^* from the binary search (4.13) and compute the RB solution $u_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu})$ from (4.18). The RB residual is

$$r_{\bar{N}}^{h\text{RB}}(v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) - a(u_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) \quad \forall v \in X; \quad (4.19)$$

the Riesz representation of the residual is denoted by $\mathcal{R}_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu})$. Our upper bound for the X -norm error $\|u(\boldsymbol{\mu}) - u_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu})\|_X$ is then given by

$$\Delta_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu}) \equiv \frac{\|\mathcal{R}_{\bar{N}}^{h\text{RB}}(\boldsymbol{\mu})\|_X}{\alpha_{\text{LB}}(\boldsymbol{\mu})}. \quad (4.20)$$

Lemma 2 now directly applies with an appropriate change of notation.

Remark 3 (the error bound as proximity function). *For any $B_l \in \mathcal{B}_l$ (associated with a nonempty model), $1 \leq l \leq L$, and any $\boldsymbol{\mu} \in \mathcal{D}$, we can derive the RB error bound associated with the RB approximation to $u(\boldsymbol{\mu})$ in the space $X_{\bar{N}, B_l}$; we denote this error bound by $\Delta_{\bar{N}, B_l}(\boldsymbol{\mu})$. As an alternative to the proximity function introduced in (4.12), we can use*

$$d_{B_l}(\boldsymbol{\mu}) = \Delta_{\bar{N}=1, B_l}(\boldsymbol{\mu}) \quad (4.21)$$

to measure the “distance” between the points $\hat{\boldsymbol{\mu}}_{B_i}$ and $\boldsymbol{\mu}$. Note that we use the error bound associated with the RB approximation for $\bar{N} = 1$ (which is simply a multiple of the snapshot associated with the anchor point) even when $\bar{N} > 1$; hence evaluation of (4.21) does not depend on \bar{N} . In section 6, we provide results with the proximity function defined both as in (4.12) and as in (4.21).

Remark 4 (multiple inner products). *The “h”-type RB approximation offers a natural way of introducing multiple X -inner products (2.4) in the computation of the dual norm of the residual for the a posteriori error bounds—we may choose a different X -inner product for each subdomain. For example, we could choose the anchor point in any subdomain to be the reference parameter, and thus define an “optimized” inner product, associated with that subdomain. With this approach, we would expect sharper error bounds and thus a better parameter domain partition (as well as, ultimately, greater online efficiency).*

To compute the dual norm of the residual we must (in the construction stage) solve a number of problems of the form (3.4) with different right-hand sides. If we solve the discrete system directly, we must invert one operator for each inner product; hence there is a computational advantage associated with only a single inner product. If we solve the discrete system iteratively, however, we can introduce individual inner products within each subdomain at very little computational penalty. In this paper, however, we have not pursued a multiple inner product approach for our numerical examples.

4.4 Offline-online decomposition

In the *offline* stage, we determine the partition of the parameter domain and construct the corresponding RB models and spaces: we execute the command $h\text{RB}(\Xi_{(1)}, \hat{\boldsymbol{\mu}}_{(1)}, \bar{N}, \epsilon_{\text{tol}}^1)$. For the purposes of this subsection, we assume a perfect binary tree; note that a perfect binary tree with K leaf nodes has $2K - 1$ nodes in total. We also assume that the cardinality of the train sample over each of the subdomains is equal to n_{train} .

The offline stage computational cost derives from several components:

1. *Snapshot truth solves.* In the case $\bar{N} \geq 2$ each node in the tree except the root node inherits one snapshot from its parent. We must thus compute \bar{N} snapshots for the model associated with the root node, and $\bar{N} - 1$ additional snapshots for each of the $2K - 2$ models associated with all nodes except the root node. Note that we retain only the basis functions associated with the K leaf nodes; we thus discard $\bar{N} - 2$ snapshots for each

intermediate (nonleaf) model. In the case $\bar{N} = 1$ we compute $\bar{N}K$ snapshots in total since the snapshot associated with an intermediate model is inherited by one of the children.

2. *RB preprocessing.* In the case $\bar{N} \geq 2$ we must compute $(Q_a\bar{N}^2 + Q_f\bar{N})$ truth inner products to form the parameter-independent “stiffness” matrices and loads (e.g., as in (3.12)) for the model associated with the root node, and $(2K - 2)(Q_a(\bar{N}^2 - 1) + Q_f(\bar{N} - 1))$ additional truth inner products in total to form the parameter-independent “stiffness” matrices and loads for the remaining $2K - 2$ models. In the case $\bar{N} = 1$ we must compute $(2K - 1)(Q_a\bar{N}^2 + Q_f\bar{N})$ truth inner products in total to form the parameter-independent “stiffness” matrices and loads.
3. *Error bound preprocessing.* In the case $\bar{N} \geq 2$ we must compute $\bar{N}Q_a + Q_f$ truth Poisson solves of the form (3.18) for the model associated with the root node, and $(2K - 2)(\bar{N} - 1)Q_a$ additional truth Poisson solves in total for the remaining $(2K - 2)$ models. We must also compute $(\bar{N}Q_a + Q_f)^2$ truth inner products of the form (3.21) related to the dual norm of the residual for the model associated with the root node, and $(2K - 2)((\bar{N}Q_a + Q_f)^2 - (Q_a + Q_f)^2)$ additional truth inner products in order to evaluate the dual norm of the residual for the $(2K - 2)$ remaining models. In the case $\bar{N} = 1$ we must compute $(2K - 1)\bar{N}Q_a + Q_f$ truth Poisson solves in total and $(2K - 1)(\bar{N}Q_a + Q_f)^2$ truth inner-products in total.
4. *Error bound evaluations.* For $\bar{N} \geq 1$ we must solve $n_{\text{train}}\bar{N}(2K - 1)$ RB systems to obtain the residual coefficients and evaluate $n_{\text{train}}\bar{N}(2K - 1)$ RB error bounds during the Greedy sampling including both the intermediate and final models. This results, to leading order, in $\mathcal{O}(n_{\text{train}}\bar{N}(2K - 1)(\bar{N}^3 + \bar{N}^2Q^2))$ operations in total.

In the case $\bar{N} \geq 2$ the combined 1–4 offline cost is thus approximately $2\bar{N}K$ truth snapshot computations, $2Q_a\bar{N}K + Q_f$ truth Poisson solves, $2K(Q_a\bar{N}^2 + Q_f\bar{N}) + 2K(\bar{N}Q_a + Q_f)^2$ truth inner products, and $\mathcal{O}(n_{\text{train}}2\bar{N}K(\bar{N}^3 + \bar{N}^2Q^2))$ operations to evaluate the error bounds. Note that the additional cost associated with the (ultimately discarded) $K - 1$ intermediate models required for the construction of the parameter domain partition is not onerous—a factor of two. In the case $\bar{N} = 1$ we retain all computed entities associated with the intermediate models during the partition procedure for the final models, and there is thus only minor additional cost (anchor point identification in 4) associated with the partition procedure.

The link between the offline and online stages is the parameter-independent data *constructed* in the offline stage and stored (permanently) for *evaluation* in the online stage. Since we retain only the data associated with the final models, the online storage for the “*h*”-type RB approximation is $Q_a K$ matrices of size $\bar{N} \times \bar{N}$ and $Q_f K$ vectors of size \bar{N} ; the online storage associated with the RB error bound is $K(\bar{N}Q_a + Q_f)^2/2$. (If we were to retain intermediate models for purposes of online adaptivity, clearly the online storage would increase; we do not consider this case further since in actual practice online adaptivity is typically pursued through the “*hp*”-approach.)

In the *online* stage, given any $\boldsymbol{\mu} \in \mathcal{D}$, we first determine the subdomain which contains $\boldsymbol{\mu}$ via the binary search (4.13) in $\mathcal{O}(\log_2 K)$ operations. Thanks to the construction-evaluation decomposition, we can then assemble and solve the corresponding system of algebraic equations in $\mathcal{O}(Q\bar{N}^2)$ and $\mathcal{O}(\bar{N}^3)$ operations, respectively, and compute the associated *a posteriori* error bound in $\mathcal{O}(\bar{N}^2 Q^2)$ operations. Note that the search (4.13) is an $\mathcal{O}(\log_2 K)$ operation only under the hypothesis that the depth of the tree associated with the partition of \mathcal{D} , L , is proportional to $\log_2 K$; we provide numerical results to support this hypothesis in section 6. We also emphasize that the efficient $\mathcal{O}(\log_2 K)$ search is a particular property of our hierarchical partition construction; if we were to partition the parameter domain based on (say) a Voronoi diagram, determination of the subdomain which contains $\boldsymbol{\mu} \in \mathcal{D}$ would be less efficient.

4.5 *A priori* theory: $\bar{N} = 1$, $P = 1$

In this section we develop an *a priori* convergence theory for an “*h*”-type RB approximation of zeroth order ($\bar{N} = 1$) in the one-parameter case ($P = 1$) when the Euclidean distance is used as the proximity function. We focus on $\bar{N} = 1$ since in fact $\bar{N} = 1$ is crucial to the “*hp*”-approach of section 5: the theory developed here demonstrates that an $\bar{N} = 1$ Greedy approach can indeed generate a reasonably efficient partition; convergence is crucial for offline and also ultimately online performance. We consider $P = 1$ for simplicity; at the conclusion of this section we provide a remark addressing (nonrigorously) the $\bar{N} > 1$ (higher “order”) and $P > 1$ cases.

For our purposes here, we do not need the Boolean indexing of the anchor points and subdomains: we shall consider Algorithm 3 after the generation of K subdomains; we relabel the K anchor points as $\hat{\boldsymbol{\mu}}'_1, \hat{\boldsymbol{\mu}}'_2, \dots, \hat{\boldsymbol{\mu}}'_K$ (numbered in the order in which they are chosen by Algorithm 3). When Algorithm 3 identifies a new anchor point (and thus subdomain), the parameter domain partition changes; we introduce mappings $I_{\tilde{K}} : \mathcal{D} \rightarrow \{1, \dots, \tilde{K}\}$, $1 \leq \tilde{K} \leq K$,

such that with \tilde{K} anchor points, for any $\boldsymbol{\mu} \in \mathcal{D}$,

$$\hat{\boldsymbol{\mu}}'_{I_{\tilde{K}}(\boldsymbol{\mu})} = \hat{\boldsymbol{\mu}}_{B^*(\boldsymbol{\mu}; \tilde{K})}, \quad (4.22)$$

where $B^*(\boldsymbol{\mu}; \tilde{K})$ is the Boolean index of the particular subdomain (among the \tilde{K} subdomains) containing $\boldsymbol{\mu}$. Below, we omit the ' for brevity.

For the purpose of this section, given \tilde{K} anchor points and corresponding subdomains, we denote by $u_{\tilde{K}}(\boldsymbol{\mu})$ the zeroth-order ($\bar{N} = 1$) “ h ”-type RB approximation for any $\boldsymbol{\mu} \in \mathcal{D}$. With the implicit mapping above, we have

$$u_{\tilde{K}}(\boldsymbol{\mu}) = \omega_{\tilde{K}}(\boldsymbol{\mu})u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})}), \quad (4.23)$$

where the coefficient $\omega_{\tilde{K}}(\boldsymbol{\mu})$ is given by the Galerkin projection as

$$\omega_{\tilde{K}}(\boldsymbol{\mu}) = \frac{f(u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})}); \boldsymbol{\mu})}{a(u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})}), u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})}); \boldsymbol{\mu})}. \quad (4.24)$$

Note that (4.23) holds since we consider $\bar{N} = 1$: a *single* RB basis function associated with each subdomain. We denote by $r_{\tilde{K}}(v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) - a(u_{\tilde{K}}(\boldsymbol{\mu}), v; \boldsymbol{\mu})$ the RB residual and let $\mathcal{R}_{\tilde{K}}(\boldsymbol{\mu}) \in X$ satisfy $(\mathcal{R}_{\tilde{K}}(\boldsymbol{\mu}), v)_X = r_{\tilde{K}}(v; \boldsymbol{\mu})$ for all $v \in X$. Our X -norm error upper bound is then written in this subsection as

$$\Delta_{\tilde{K}}(\boldsymbol{\mu}) = \frac{\|\mathcal{R}_{\tilde{K}}(\boldsymbol{\mu})\|_X}{\alpha_{\text{LB}}(\boldsymbol{\mu})}, \quad (4.25)$$

which is simply a specialization of (4.20).

We need two further preliminary results. First, it is clear from Cea’s lemma (with respect to the X -norm), (2.9), and (2.8) that for any \tilde{K} , $1 \leq \tilde{K} \leq K$, and any $\boldsymbol{\mu} \in \mathcal{D}$,

$$\|u(\boldsymbol{\mu}) - u_{\tilde{K}}(\boldsymbol{\mu})\|_X \leq \frac{\bar{\gamma}}{\underline{\alpha}} \|u(\boldsymbol{\mu}) - u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})})\|_X, \quad (4.26)$$

since $u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\boldsymbol{\mu})})$ is a particular member of the (one-dimensional) RB space. Second, from (3.7) of Lemma 2, we obtain for any \tilde{K} , $1 \leq \tilde{K} \leq K$, and any $\boldsymbol{\mu} \in \mathcal{D}$,

$$\Delta_{\tilde{K}}(\boldsymbol{\mu}) \leq \frac{\bar{\gamma}}{\underline{\alpha}} \|u(\boldsymbol{\mu}) - u_{\tilde{K}}(\boldsymbol{\mu})\|_X, \quad 1 \leq \tilde{K} \leq K. \quad (4.27)$$

We can now state the following proposition.

Proposition 1 (convergence in the case $\bar{N} = 1, P = 1$). *The “h”-type RB approximation terminates for finite $K(\epsilon_{\text{tol}}^1)$ subdomains. Further, the convergence is first order in the sense that*

$$K(\epsilon_{\text{tol}}^1) \leq \max \left\{ 1, \frac{C}{\epsilon_{\text{tol}}^1} \right\} \quad (4.28)$$

for a constant C given by

$$C = \frac{2\bar{\gamma}^2 \tilde{C} |\mathcal{D}|}{\underline{\alpha}^2}, \quad (4.29)$$

where $\tilde{C} = (\tilde{c}_1 \max_{\boldsymbol{\mu} \in \mathcal{D}} \|f(\cdot; \boldsymbol{\mu})\|_{X'} + \underline{\alpha} \tilde{c}_2) / \underline{\alpha}^2$ is the constant developed in Lemma 1 and $|\mathcal{D}|$ is the length of $\mathcal{D} \subset \mathbb{R}$.

Proof. Algorithm 3 provides a sequence of anchor points $\hat{\boldsymbol{\mu}}_1, \dots, \hat{\boldsymbol{\mu}}_K$ for $K \geq 1$. We have by construction of our algorithm that either $K = 1$ or $K > 1$ and

$$\epsilon_{\text{tol}}^1 < \Delta_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1}), \quad 1 \leq \tilde{K} \leq K - 1. \quad (4.30)$$

In the former case the proof is complete; we henceforth consider the latter case.

We deduce from (4.27), (4.26), and Lemma 1, respectively, that

$$\Delta_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1}) \leq \frac{\bar{\gamma}}{\underline{\alpha}} \|u(\hat{\boldsymbol{\mu}}_{\tilde{K}+1}) - u_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})\|_X \quad (4.31)$$

$$\leq \frac{\bar{\gamma}^2}{\underline{\alpha}^2} \|u(\hat{\boldsymbol{\mu}}_{\tilde{K}+1}) - u(\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})})\|_X \quad (4.32)$$

$$\leq \frac{\bar{\gamma}^2}{\underline{\alpha}^2} \tilde{C} |\hat{\boldsymbol{\mu}}_{\tilde{K}+1} - \hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}| \quad (4.33)$$

for $1 \leq \tilde{K} \leq K - 1$; hence from (4.30)

$$|\hat{\boldsymbol{\mu}}_{\tilde{K}+1} - \hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}| > \frac{\underline{\alpha}^2 \epsilon_{\text{tol}}^1}{\bar{\gamma}^2 \tilde{C}} \quad (4.34)$$

for $1 \leq \tilde{K} \leq K - 1$.

For \tilde{K} , $1 \leq \tilde{K} \leq K - 1$, the algorithm selects the next anchor point $\hat{\boldsymbol{\mu}}_{\tilde{K}+1}$ and the intermediate subdomain associated with anchor point number $I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})$ is divided into two new subdomains. It is clear that the length of each of the two new subdomains is at least as large as half the distance between the new anchor

point $\hat{\boldsymbol{\mu}}_{\tilde{K}+1}$ and anchor point $\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}$, namely, $|\hat{\boldsymbol{\mu}}_{\tilde{K}+1} - \hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}|/2$ (recall that the distance function is the Euclidean distance and that we consider $P = 1$ parameter). Let δ_k^K denote the length of the subdomain associated with anchor point $\hat{\boldsymbol{\mu}}_k$, $1 \leq k \leq K$. We note that each of the K subdomains generated by Algorithm 3 results from the splitting of the intermediate subdomain associated with anchor point $\hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}$ for some $\tilde{K} \in \{1, \dots, K-1\}$; hence for $1 \leq k \leq K$, there exists a $\tilde{K} \in \{1, \dots, K-1\}$ such that

$$\delta_k^K \geq |\hat{\boldsymbol{\mu}}_{\tilde{K}+1} - \hat{\boldsymbol{\mu}}_{I_{\tilde{K}}(\hat{\boldsymbol{\mu}}_{\tilde{K}+1})}|/2, \quad (4.35)$$

and thus by (4.34)

$$\underline{\delta}^K \equiv \min_{1 \leq k \leq K} \delta_k^K > \frac{\underline{\alpha}^2 \epsilon_{\text{tol}}^1}{2\bar{\gamma}^2 \tilde{C}}. \quad (4.36)$$

Note that $\underline{\delta}^K$ is not the smallest distance between two anchor points: rather, it is the smallest length of any of the K subdomains.

Let $|\mathcal{D}|$ denote the length of \mathcal{D} . With K subdomains, it is clear that $K\underline{\delta}^K \leq |\mathcal{D}|$. We now assume $K > C/\epsilon_{\text{tol}}^1$. From (4.36) it then follows that

$$K\underline{\delta}^K > \frac{C}{\epsilon_{\text{tol}}^1} \underline{\delta}^K \geq \left(\frac{2\bar{\gamma}^2 \tilde{C} |\mathcal{D}|}{\underline{\alpha}^2 \epsilon_{\text{tol}}^1} \right) \left(\frac{\underline{\alpha}^2 \epsilon_{\text{tol}}^1}{2\bar{\gamma}^2 \tilde{C}} \right) = |\mathcal{D}|, \quad (4.37)$$

which is clearly false. We have thus reached a contradiction: the “ h ”-type RB approximation cannot generate a sequence of anchor points $\hat{\boldsymbol{\mu}}_1, \dots, \hat{\boldsymbol{\mu}}_K$ for $K > C/\epsilon_{\text{tol}}^1$; thus the algorithm must terminate for $1 \leq K \leq C/\epsilon_{\text{tol}}^1$ subdomains. \square

Remark 5 (convergence in the case $\bar{N} \geq 1$, $P \geq 1$). *We first recall a polynomial approximation result. Consider approximation of a sufficiently smooth function on a bounded domain in \mathbb{R}^P by piecewise polynomial interpolation of order p over K subdomains: we expect the error to decrease as $(1/K)^{(p+1)/P}$, or as $(1/K)^{(\bar{N}^{1/P})/P}$ if we associate $\bar{N} = (p+1)^P$ degrees of freedom to each subdomain (consistent with our earlier definition of “order”).*

In the zeroth-order multiparameter case ($\bar{N} = 1$, $P > 1$) we expect (but do not prove) that our method converges for

$$K < \max \left\{ 1, \frac{C}{(\epsilon_{\text{tol}}^1)^P} \right\} \quad (4.38)$$

subdomains for some positive constant C . This poor convergence for $P \gg 1$ suggests the advantage and indeed necessity of “ p ”-convergence [25] or “ hp ”-convergence rather than solely “ h ”-convergence. Next, in the higher order, one-parameter case ($\bar{N} > 1$, $P = 1$), we might expect convergence

$$K < \max\left\{1, \frac{C}{(\epsilon_{\text{tol}}^1)^{\frac{1}{\bar{N}}}}\right\} \quad (4.39)$$

for some positive constant C . Finally, in the general case ($\bar{N} \geq 1$, $P \geq 1$), we might expect convergence

$$K < \max\left\{1, \frac{C}{(\epsilon_{\text{tol}}^1)^{\frac{P}{\bar{N}^{1/P}}}}\right\}. \quad (4.40)$$

We shall consider these heuristic arguments again in the context of numerical results.

Note that our bound (4.28) and estimates (4.39) and (4.40) should capture the correct order, but of course the constant will be very pessimistic: by design, the Greedy algorithm should adapt the sample to best accommodate local variations.

Remark 6 (a sharper convergence result). We show in the appendix that the constant C in Proposition 1 can be improved to $C_{\text{im}} = 2\tilde{C}|\mathcal{D}|(1 + \bar{\gamma}/\underline{\alpha})$ by an approach that, instead of (4.31)–(4.33), directly considers the equation for the Riesz representation of the residual. However, the approach above is more general and applicable in other (e.g., interpolation [11]) contexts.

5 The “ hp ”-type RB method

With the “ hp ”-type RB method, we combine the “ h ”- and “ p ”-type methods: we first construct a partition of the parameter domain with “ h ”-refinement; we then compute independent approximation spaces restricted to each parameter subdomain with “ p ”-refinement—in general, the approximation spaces will have different dimensions.

5.1 Approximation

The parameter domain partition is first constructed by an $\bar{N} = 1$ “ h ”-type approximation for a prescribed error bound tolerance ϵ_{tol}^1 . We first construct the

initial train sample $\Xi_{(1)} \subset \mathcal{D}$, choose an initial parameter anchor $\hat{\boldsymbol{\mu}}_{(1)} \in \mathcal{D}$, and specify ϵ_{tol}^1 ; we then execute Algorithm 3, $h\text{RB}(\Xi_{(1)}, \hat{\boldsymbol{\mu}}_{(1)}, 1, \epsilon_{\text{tol}}^1)$. The output from $h\text{RB}(\Xi_{(1)}, \hat{\boldsymbol{\mu}}_{(1)}, 1, \epsilon_{\text{tol}}^1)$ is K subdomains associated with the K leaf nodes of the binary tree. Each subdomain has an associated one-parameter model and a one-dimensional approximation space; we denote by B^1, \dots, B^K the K associated Boolean indices. We also store the train sample over each of the final subdomains.

We now append additional basis functions to each approximation space by a standard “ p ”-type procedure over each train sample: we specify the maximum RB space dimension $\tilde{N}_{\max, B^k} = \tilde{N}_{\max} > \tilde{N}$, $1 \leq k \leq K$; we specify a new error bound tolerance $\epsilon_{\text{tol}}^2 < \epsilon_{\text{tol}}^1$; we then execute Algorithm 2, Greedy²($\Xi_{B^k}, \boldsymbol{\mu}_{1, B^k}, \epsilon_{\text{tol}}^2, \tilde{N}_{\max, B^k}$) for $1 \leq k \leq K$. (Note that we must replace the “ h ”-type error bound $\Delta_N^{h\text{RB}}$ in Algorithm 2 by the “ hp ”-type error bound $\Delta_N^{hp\text{RB}}$, which we introduce shortly.)

The final output is thus K RB approximation spaces X_{N_{\max, B^k}, B^k} and associated models $\mathcal{M}_{N_{\max, B^k}, B^k}$, $1 \leq k \leq K$. Note that N_{\max, B^k} is in general different for different k since the error bound tolerance ϵ_{tol}^2 might be satisfied by the different approximation spaces over the different train samples with different numbers of basis functions; we define in the “ hp ” case $N_{\max} = \max_{k=1, \dots, K} N_{\max, B^k}$.

Finally, we may now define the “ hp ”-type RB approximation. First, given any $\boldsymbol{\mu} \in \mathcal{D}$, we determine the subdomain $\mathcal{V}_{B_i^*}$ containing $\boldsymbol{\mu}$ from the search (4.13). Then, given $1 \leq N \leq N_{\max}$, we find $u_N^{hp\text{RB}}(\boldsymbol{\mu}) \in X_{N^*, B_i^*}$ such that

$$a(u_N^{hp\text{RB}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) \quad \forall v \in X_{N^*, B_i^*}, \quad (5.1)$$

where $N^* \equiv \min\{N, N_{\max, B_i^*}\}$. (Note that B_i^* and thus N^* depend on $\boldsymbol{\mu}$.)

5.2 *A posteriori* error estimation

We shall require some new notation for the “ hp ”-type *a posteriori* error bound.

Given any $\boldsymbol{\mu} \in \mathcal{D}$ and a partition of \mathcal{D} into subdomains, we determine B_i^* from the binary search (4.13) and compute the RB solution $u_N^{hp\text{RB}}(\boldsymbol{\mu})$ from (5.1). The RB residual is

$$r_N^{hp\text{RB}}(v; \boldsymbol{\mu}) = f(v; \boldsymbol{\mu}) - a(u_N^{hp\text{RB}}(\boldsymbol{\mu}), v; \boldsymbol{\mu}) \quad \forall v \in X; \quad (5.2)$$

the Riesz representation of the residual is denoted by $\mathcal{R}_N^{hp\text{RB}}(\boldsymbol{\mu})$. Our upper

bound for the X -norm error $\|u(\boldsymbol{\mu}) - u_N^{hpRB}(\boldsymbol{\mu})\|_X$ is then given by

$$\Delta_N^{hpRB}(\boldsymbol{\mu}) \equiv \frac{\|\mathcal{R}_N^{hpRB}(\boldsymbol{\mu})\|_X}{\alpha_{LB}(\boldsymbol{\mu})}. \quad (5.3)$$

Lemma 2 now directly applies with an appropriate change of notation.

5.3 Offline-online decomposition

In the *offline* stage, we determine the partition of the parameter domain and construct the corresponding RB models and spaces as discussed above. For the purposes of this subsection, we assume a perfect binary tree. We also assume that the cardinality of the train sample over each of the subdomains is equal to n_{train} . It is crucial to note that, since the initial “ h ”-refinement is performed for $\bar{N} = 1$, we reuse all computed entities in the later “ p ”-type stage, and there is thus only minor additional cost associated with the partition procedure.

The offline cost derives from several components.

1. *Snapshot truth solves.* We must compute (at most) $N_{\text{max}}K$ snapshots associated with the partition *and* final approximation spaces.
2. *RB preprocessing.* We must compute (at most) $K(Q_a N_{\text{max}}^2 + Q_f N_{\text{max}})$ truth inner products to form the parameter-independent “stiffness” matrices and loads (e.g., as in (3.12)) for the partition and final models.
3. *Error bound preprocessing.* We must compute (at most) $KN_{\text{max}}Q_a + Q_f$ truth Poisson solves of the form (3.18) for the partition and final models. We must also compute (at most) $K(N_{\text{max}}Q_a + Q_f)^2$ truth inner products of the form (3.21) related to the dual norm of the residual associated with the partition and final models.
4. *Error bound evaluations.* We must solve $n_{\text{train}}(K - 1)$ RB systems (of size $\bar{N} = 1$) to obtain the residual coefficients and evaluate $n_{\text{train}}(K - 1)$ RB error bounds (for $\bar{N} = 1$) during the Greedy² sampling for the “ h ”-refinement partition process. We must also solve (at most) $n_{\text{train}}N_{\text{max}}K$ RB systems to obtain the residual coefficients and evaluate $n_{\text{train}}N_{\text{max}}K$ RB error bounds during the Greedy² sampling for the final models. This results, to leading order, in $\mathcal{O}(n_{\text{train}}N_{\text{max}}K(N_{\text{max}}^3 + N_{\text{max}}^2Q^2))$ operations in total.

The combined 1–4 offline cost is thus $N_{\max}K$ truth snapshot computations, $N_{\max}KQ_a + Q_f$ truth Poisson solves, $K(Q_a N_{\max}^2 + Q_f N_{\max}) + K(N_{\max}Q_a + Q_f)^2$ truth inner products, and $\mathcal{O}(n_{\text{train}}N_{\max}K(N_{\max}^3 + N_{\max}^2Q^2))$ operations to evaluate the error bounds.

For each model, we must construct and retain the parameter-independent data necessary to accommodate the efficient evaluation stage for the RB approximation and the associated *a posteriori* error bound, as discussed in section 3.3 for the standard RB method. The online (permanent) storage requirement associated with the RB approximation is $Q_a K$ matrices of maximum size $N_{\max} \times N_{\max}$ and Q_f vectors of maximum size N_{\max} ; the online storage associated with the RB error bound is $K(N_{\max}Q_a + Q_f)^2/2$.

In the *online* stage, given any $\boldsymbol{\mu} \in \mathcal{D}$, we first determine the subdomain containing $\boldsymbol{\mu}$ via the binary search (4.13) in $\mathcal{O}(\log_2 K)$ operations. (Recall that we presume here a perfect binary tree.) Thanks to the construction-evaluation decomposition, we can then, given $1 \leq N \leq N_{\max}$, assemble and solve the corresponding system of algebraic equations in $\mathcal{O}(QN^2)$ and $\mathcal{O}(N^3)$ operations, respectively, and compute the associated RB error bound in $\mathcal{O}(N^2Q^2)$ operations.

6 A convection-diffusion model problem

6.1 Formulation and truth discretization

We now apply the “*p*”-, “*h*”-, and “*hp*”-type RB methods to a steady convection-diffusion model problem parametrized by the angle and magnitude of the prescribed velocity field: Let $\boldsymbol{\mu} = (\mu_1, \mu_2)$ and define $\mathbf{V}(\boldsymbol{\mu}) = [\mu_2 \cos \mu_1, \mu_2 \sin \mu_1]^T$. The governing equations for the exact field variable $u^e(\boldsymbol{\mu})$ are

$$-\nabla^2 u^e(\boldsymbol{\mu}) + \mathbf{V}(\boldsymbol{\mu}) \cdot \nabla u^e(\boldsymbol{\mu}) = 10 \quad \text{in } \Omega, \quad (6.1)$$

$$u^e(\boldsymbol{\mu}) = 0 \quad \text{on } \partial\Omega. \quad (6.2)$$

The physical domain is $\Omega = \{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leq 2\}$ and $\partial\Omega$ is the boundary of Ω .

We next define for all $w, v \in X^e \equiv X^e(\Omega) \equiv H_0^1(\Omega)$ the parametrized bilinear

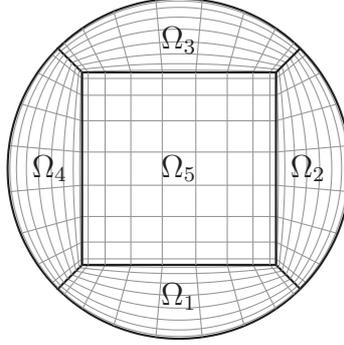


Figure 3: The circular physical domain partitioned into five spectral elements.

form

$$\begin{aligned} a(w, v; \boldsymbol{\mu}) &\equiv \int_{\Omega} \nabla w \cdot \nabla v \, d\Omega + \int_{\Omega} (\mathbf{V}(\boldsymbol{\mu}) \cdot \nabla w) v \, d\Omega \\ &\equiv \int_{\Omega} \nabla w \cdot \nabla v \, d\Omega + \mu_2 \cos \mu_1 \int_{\Omega} \frac{\partial w}{\partial x} v \, d\Omega + \mu_2 \sin \mu_1 \int_{\Omega} \frac{\partial w}{\partial y} v \, d\Omega \end{aligned} \quad (6.3)$$

and the linear functional

$$f(v) \equiv f(v; \boldsymbol{\mu}) \equiv 10 \int_{\Omega} v \, d\Omega; \quad (6.4)$$

thus (2.1) obtains for $Q_a = 3$ and $Q_f = 1$. We can then state the exact problem in the standard variational form: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u^e \in X^e$ such that

$$a(u^e(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v) \quad \forall v \in X^e. \quad (6.5)$$

Note that for this particular problem, $a_s(w, v; \boldsymbol{\mu}) = \int_{\Omega} \nabla w \cdot \nabla v \, d\Omega$ is parameter-independent; thus $a(v, v; \boldsymbol{\mu}) = \|v\|_X^2$ for all $v \in X^e$ and we may choose $\alpha_{\text{LB}} \equiv 1$ as the coercivity lower bound.

Next, we introduce a truth spectral element space $X \equiv X^{\mathcal{N}}(\Omega) \subset X^e(\Omega)$ of dimension $\mathcal{N} = 481$ based on five spectral elements of order ten: we introduce a computational domain $\hat{\Omega} = (-1, 1)^2$ and standard transfinite mappings $\mathcal{F}_i : \hat{\Omega} \rightarrow \Omega_i$, $1 \leq i \leq 5$ [12]; we then define

$$X \equiv X^{\mathcal{N}}(\Omega) = \{v \in H_0^1(\Omega) : v|_{\Omega_i} \circ \mathcal{F}_i \in \mathbb{P}^{10}(\hat{\Omega}), 1 \leq i \leq 5\}, \quad (6.6)$$

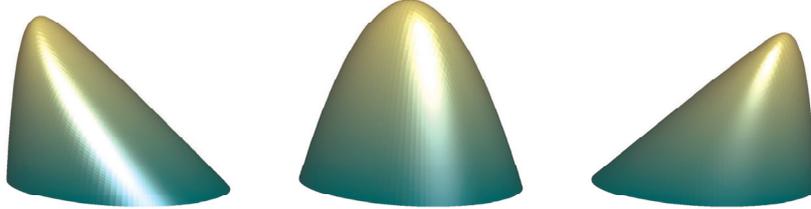


Figure 4: Solutions to (6.7) for different parameter values $\boldsymbol{\mu} = (\pi, 10)$ (left), $\boldsymbol{\mu} = (0, 0)$ (middle), and $\boldsymbol{\mu} = (0, 10)$ (right).

where $\mathbb{P}^{10}(\hat{\Omega})$ denotes the space of polynomials of degree 10 (in each spatial direction) over $\hat{\Omega}$. The partition of the physical domain into five spectral elements is illustrated in Figure 3. The truth discretization of (6.5) reads as follows: Given any $\boldsymbol{\mu} \in \mathcal{D}$, find $u(\boldsymbol{\mu}) \in X$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}) = f(v) \quad \forall v \in X. \quad (6.7)$$

In Figure 4, we plot the solution of (6.7) for three different parameter values. Clearly, the three solutions have very different structures—this particular problem is thus a good candidate for “ hp ” treatment.

We define three parameter domains,

$$\mathcal{D}_I \equiv \{0\} \times [0, 10], \quad \mathcal{D}_{II} \equiv [0, \pi] \times \{10\}, \quad \mathcal{D}_{III} \equiv [0, \pi] \times [0, 10]; \quad (6.8)$$

we shall thus consider $P = 1$ (\mathcal{D}_I or \mathcal{D}_{II}) or $P = 2$ (\mathcal{D}_{III}) parameters.

Admittedly, the computational benefit of the RB approximation for this particular problem is minimal since the truth approximation space is of rather low dimension $\mathcal{N} = 481$. However, our problem here is a good vehicle for exposition of the “ p ”-type, “ h ”-type, and “ hp ”-type RB methods and is amenable to extensive theoretical and experimental analysis. In [9] we apply the “ hp ”-type RB method to problems that require and demonstrate the speedup provided by the RB approximation.

6.2 “ p ”-type RB approximation results

In this section, we present the standard (“ p ”-type) RB convergence results for our model problem.

We introduce uniformly distributed random train samples $\Xi_I \subset \mathcal{D}_I$, $\Xi_{II} \subset \mathcal{D}_{II}$, and $\Xi_{III} \subset \mathcal{D}_{III}$ of size 10^3 , 10^3 , and 10^4 , respectively. We recall that $\epsilon_N =$

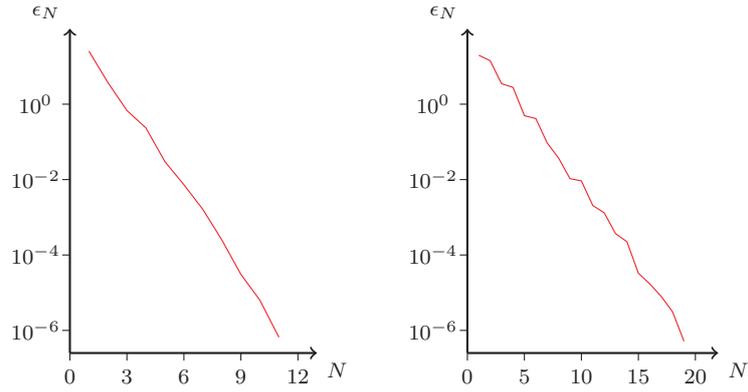


Figure 5: Standard RB (“ p ”-type) convergence results: ϵ_N as a function of N for the one-parameter cases $\mathcal{D} = \mathcal{D}_I$ (left) and $\mathcal{D} = \mathcal{D}_{II}$ (right).

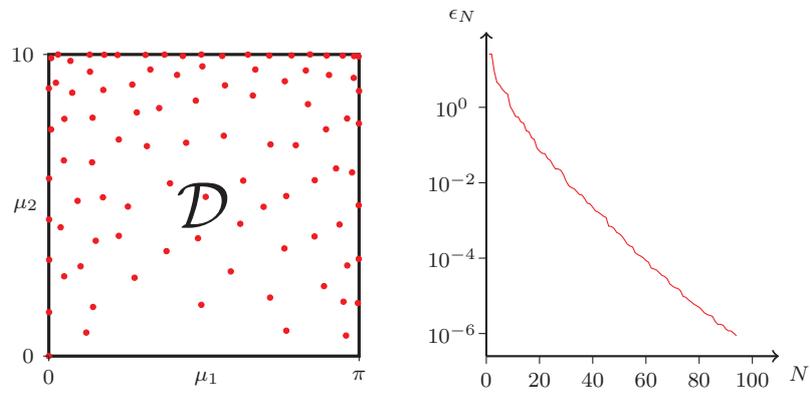


Figure 6: Greedy parameter choices (left) and associated standard RB (“ p ”-type) convergence results (right) for the two-parameter case $\mathcal{D} = \mathcal{D}_{III}$.

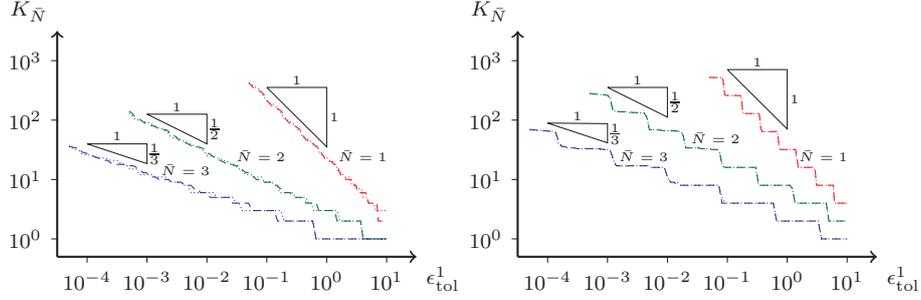


Figure 7: “ h ”-type RB convergence results: $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ for $\bar{N} = 1$, $\bar{N} = 2$, and $\bar{N} = 3$ for the one-parameter cases $\mathcal{D} = \mathcal{D}_I$ (left) and $\mathcal{D} = \mathcal{D}_{II}$ (right). Both the Euclidean distance (dotted lines) and the *a posteriori* error bound (dashed lines) are considered for the proximity function.

$\max_{\mu \in \Xi} \Delta_N(\mu)$ is the maximum X -norm error bound over the train sample associated with the space X_N . In Figure 5 we plot ϵ_N as a function of N for the two one-parameter cases $\mathcal{D} = \mathcal{D}_I$ and $\mathcal{D} = \mathcal{D}_{II}$: we note that N can be quite small even for $\epsilon_N \approx 10^{-6}$. In Figure 6 (right) we plot ϵ_N for the two-parameter case $\mathcal{D} = \mathcal{D}_{III}$. The quite poor convergence of the “ p ”-type RB is not surprising given the very different solution structures obtained for different parameter values; variations in μ_1 are particularly difficult to resolve—as indicated by the slower convergence for the case $\mathcal{D} = \mathcal{D}_{II}$ in Figure 5 (right)—due to the “movement” of the boundary layer. In Figure 6 (left) we present the parameters chosen by the Greedy¹ algorithm: the points are clearly denser for larger velocities, which yield thinner boundary layers.

6.3 “ h ”-type RB approximation results

We now present convergence results for equi-order “ h ”-type RB approximation; the dimension of the approximation spaces is thus fixed. The convergence results are obtained by first specifying the desired tolerance ϵ_{tol}^1 as well as the RB space dimension \bar{N} , the initial train sample $\Xi_{(1)}$, and the initial anchor point $\hat{\mu}_{(1)}$; we then perform $h\text{RB}(\Xi_{(1)}, \hat{\mu}_{(1)}, \bar{N}, \epsilon_{\text{tol}}^1)$. Given \bar{N} , we let $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ denote the number of subdomains in the partition for specified ϵ_{tol}^1 .

We start with the one-parameter cases $\mathcal{D} = \mathcal{D}_I$ and $\mathcal{D} = \mathcal{D}_{II}$. In both cases, the initial train samples consist of 100 random points, and the initial anchor point is $\hat{\mu}_{(1)} = (0, 0)$. In Figure 7 we present $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ for $\bar{N} = 1, 2, 3$ for each

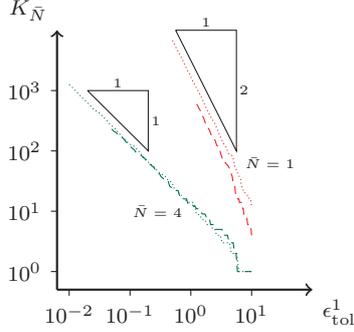


Figure 8: “ h ”-type RB convergence results: $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ for $\bar{N} = 1$ and $\bar{N} = 4$ for the two-parameter case $\mathcal{D} = \mathcal{D}_{\text{III}}$. Both the Euclidean distance (dotted lines) and the *a posteriori* error bound (dashed lines) are considered for the proximity function.

of the two cases. The proximity function is either $d_{B_i}(\boldsymbol{\mu}) = \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}_{B_i}\|_2$ (dotted lines) or $d_{B_i}(\boldsymbol{\mu}) = \Delta_{\bar{N}=1, B_i}(\boldsymbol{\mu})$ (dashed lines): we observe that the choice of the proximity function has little impact on the results. We indicate the slopes for first, second, and third order convergence: for the $\bar{N} = 1$ approximation, the convergence rates are in good agreement with the theoretical result (4.28); for the $\bar{N} > 1$ approximations, the convergence is approximately \bar{N} th order and hence in agreement with our conjecture (4.38). (We recall that, here, a steeper slope implies slower convergence.)

We next consider the two-parameter case $\mathcal{D} = \mathcal{D}_{\text{III}}$. The initial train sample $\Xi_{(1)}$ consists of 10^3 random points, and the initial anchor point is $\hat{\boldsymbol{\mu}}_{(1)} = (0, 0)$. In Figure 8 we present $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ for $\bar{N} = 1$ and $\bar{N} = 4$. The proximity function is either $d_{B_i}(\boldsymbol{\mu}) = \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}_{B_i}\|_2$ (dotted lines) or $d_{B_i}(\boldsymbol{\mu}) = \Delta_{\bar{N}=1, B_i}(\boldsymbol{\mu})$ (dashed lines): now the choice of the proximity function has some, but very slight, impact on the results but only for the $\bar{N} = 1$ approximation. It is clear from the slopes provided that we achieve roughly $K_1 \sim (\epsilon_{\text{tol}}^1)^{-2}$ and $K_4 \sim (\epsilon_{\text{tol}}^1)^{-1}$, as expected from our conjectures (4.39) and (4.40).

Finally, we empirically examine the depth of the associated binary trees. Ideally, we would like the relative tree depth (4.14) to be a constant close to unity; for K subdomains the search (4.13) is in this case an efficient $\log_2 K$ operations binary search. In Figure 9 we plot the relative tree depth against the number of subdomains for the $\bar{N} = 1$ approximation for each of our three parametrizations. (Note that the scatter in the plots is induced by the range of ϵ_{tol}^1 considered.) Although from these results it is difficult to reach general

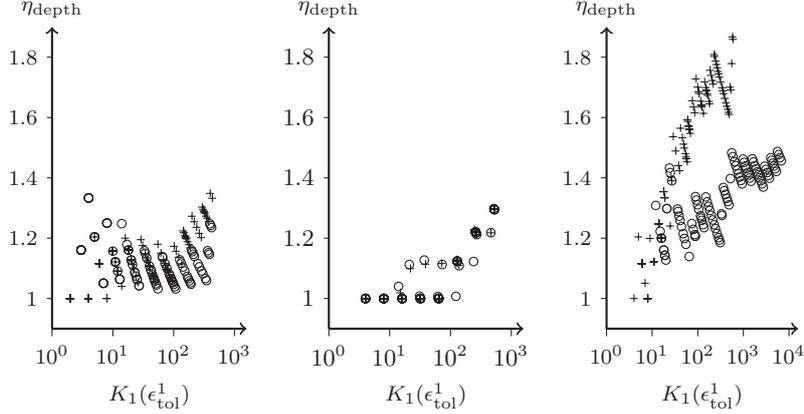


Figure 9: Relative tree depths η_{depth} as functions of the number of subdomains (leaf nodes) $K_{\bar{N}}(\epsilon_{\text{tol}}^1)$ for the $\bar{N} = 1$ “ h ”-type approximation for each parametrization $\mathcal{D} = \mathcal{D}_I$ (left), $\mathcal{D} = \mathcal{D}_{II}$ (middle), and $\mathcal{D} = \mathcal{D}_{III}$ (right). Both the Euclidean distance (\circ) and the *a posteriori* error bound ($+$) are considered for the proximity function.

conclusions, the relative tree depths are all fairly close to unity and increase with increasing K only very modestly even for $1 \leq K \leq 10^4$.

6.4 “ hp ”-type approximation results

We now present convergence results for an “ hp ”-type RB approximation. For a partition with K subdomains, let Ξ denote the union of the associated K train samples; we then define $\epsilon_N^{hpRB} \equiv \max_{\mu \in \Xi} \Delta_N^{hpRB}(\mu)$.

We start with the one-parameter cases $\mathcal{D} = \mathcal{D}_I$ and $\mathcal{D} = \mathcal{D}_{II}$. The initial train sample consists of 100 random points, and the initial anchor point is $\hat{\mu}_{(1)} = (0, 0)$. We use $d_{B_i}(\mu) = \|\mu - \hat{\mu}_{B_i}\|_2$ as the proximity function. For the case $\mathcal{D} = \mathcal{D}_I$, we specify $\epsilon_{\text{tol}}^1 = 5$ and $\epsilon_{\text{tol}}^1 = 0.1$, for which we obtain $K = 4$ and $K = 211$ subdomains, respectively; for the case $\mathcal{D} = \mathcal{D}_{II}$, we specify $\epsilon_{\text{tol}}^1 = 5$ and $\epsilon_{\text{tol}}^1 = 0.1$, for which we obtain $K = 8$ and $K = 260$ subdomains, respectively. In Figure 10 we plot ϵ_N^{hpRB} as functions of N for each of the two parametrizations. *Given* any error bound tolerance, we note a significant reduction in the required approximation space dimension (in any subdomain) when compared to a standard RB ($K = 1$) approximation. Of course, the *total* number of snapshots NK (for any given tolerance) will *increase* with K : greater

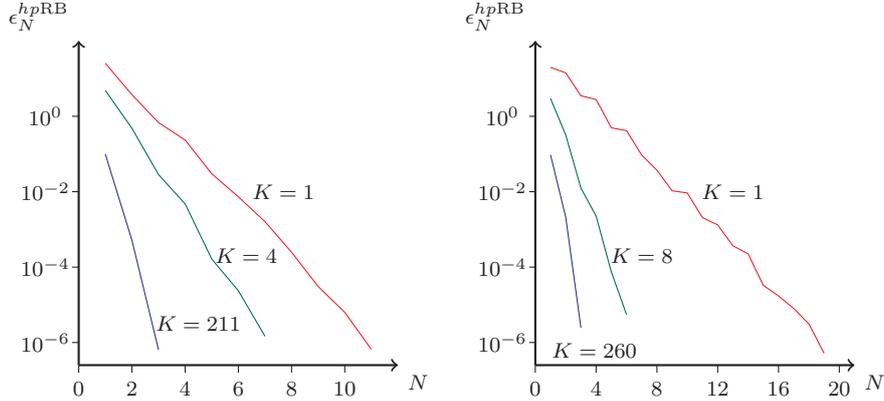


Figure 10: “ hp ”-type RB convergence results: ϵ_N^{hpRB} as a function of N for the one-parameter cases $\mathcal{D} = \mathcal{D}_I$ (left) and $\mathcal{D} = \mathcal{D}_{II}$ (right).

suitability of local snapshots does not compensate for lower order in terms of global approximation properties.

We next consider the two-parameter case $\mathcal{D} = \mathcal{D}_{III}$. We use $d_{B_i}(\boldsymbol{\mu}) = \|\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}_{B_i}\|_2$ as the proximity function. The initial train sample consists of 10^3 random points, and the initial anchor point is $\hat{\boldsymbol{\mu}}_{(1)} = (0, 0)$. In Figure 11 we show partitions of the parameter domain for specified $\epsilon_{tol}^1 = 5$ and $\epsilon_{tol}^1 = 2$, for which we obtain $K = 72$ and $K = 417$ subdomains, respectively. We note—similarly to the “ p ”-type Greedy² parameter choices in Figure 5(left)—that the subdomains are smaller for larger velocities. In Figure 12, we plot for each of the two partitions in Figure 11 the maximum error bound ϵ_N^{hpRB} as a function of N ; we include the results for the standard RB approximation (“ p ”-type or “ hp ”-type with $K = 1$) as well. Again, the local order reduction is significant.

In Tables 1 and 2 we summarize for $K = 72$ and $K = 417$ subdomains, respectively, the offline and online performance of the “ hp ” approach *relative* to that of the standard RB method. For given tolerances ϵ_{tol}^2 , we report in the four rows of the tables the relative number of truth solves, the relative number of operations for online evaluation of the RB approximation, the relative number of operations for online evaluation of the RB error bound, and relative online storage, respectively. The reported values are based on the theoretical operation count and storage, which we recall here. For N basis functions and

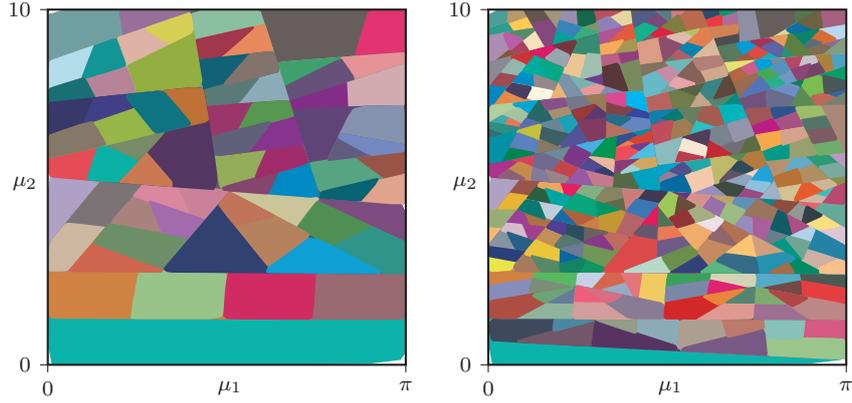


Figure 11: Parameter domain partitions for the case $\mathcal{D} = \mathcal{D}_{\text{III}}$. The number of subdomains is $K(\epsilon_{\text{tol}}^1) = 72$ for $\epsilon_{\text{tol}}^1 = 5$ (left) and $K(\epsilon_{\text{tol}}^1) = 417$ for $\epsilon_{\text{tol}}^1 = 2$ (right).

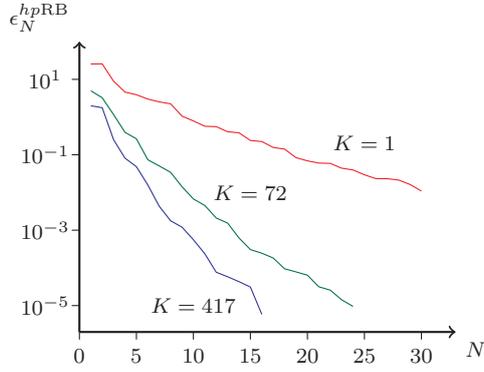


Figure 12: Convergence of “ hp ”-type RB for the two-parameter case $\mathcal{D} = \mathcal{D}_{\text{III}}$.

	$\epsilon_{\text{tol}}^2 = 10^{-2}$	$\epsilon_{\text{tol}}^2 = 10^{-3}$	$\epsilon_{\text{tol}}^2 = 10^{-4}$
Offline truth solve relative cost	1.66E+1	1.57E+1	1.59E+1
Online RB solution relative cost	3.35E-2	2.82E-2	2.70E-2
Online RB error bound relative cost	1.09E-1	9.57E-2	9.23E-2
Online relative storage	3.96E+0	3.51E+0	3.58E+0

Table 1: Operation count and storage requirement for the “ hp ”-type RB with $K = 72$ relative to that of the standard RB ($K = 1$) for the two-parameter case $\mathcal{D} = \mathcal{D}_{\text{III}}$.

	$\epsilon_{\text{tol}}^2 = 10^{-2}$	$\epsilon_{\text{tol}}^2 = 10^{-3}$	$\epsilon_{\text{tol}}^2 = 10^{-4}$
Offline truth solve relative cost	6.58E+1	6.19E+1	6.33E+1
Online RB solution relative cost	1.15E-2	1.03E-2	8.00E-3
Online RB error bound relative cost	5.48E-2	4.97E-2	4.18E-2
Online relative storage	1.08E+1	9.39E+0	9.81E+0

Table 2: Operation count and storage requirement for the “ hp ”-type RB with $K = 417$ relative to that of the standard RB ($K = 1$) for the two-parameter case $\mathcal{D} = \mathcal{D}_{\text{III}}$.

$K \geq 1$ subdomains the number of offline truth solves is $KN(1 + Q_a) + Q_f$.² (We assume roughly equal computation times for the convection-diffusion and Poisson solves.) The online operation count is roughly $\mathcal{O}(N^3)$ for the RB solution and $\mathcal{O}((Q_a N + Q_f)^2)$ for the RB error bound; we neglect the $\mathcal{O}(QN^2)$ cost of forming the RB system and the $\mathcal{O}(\log_2 K)$ cost of finding the correct subdomain via the binary search. The online (permanent) storage requirement is dominated by the $\mathcal{O}(KQ^2N^2)$ data required for the RB error bounds.

Admittedly, the “ hp ”-approach requires more truth solves—a larger offline cost—than the standard method. However, the online computational savings are significant: in our example with $K = 72$ subdomains the online cost relative to that of the standard RB method is about three percent for the RB solution and about ten percent for the RB error bound; in our example with $K = 417$ subdomains the online relative cost is only about one percent for the RB solution and about five percent for the RB error bound. (Typically the contributions to the total online cost from RB solution and RB error bound computation are

²We assume that the truth solves (including both the snapshot computation and the Poisson solves related to the dual norm of the residual) constitute the most expensive part of the total offline cost. In fact, this assumption *favours* the standard “ p ”-type RB since the error bound sampling is superlinear in N and thus scales more advantageously for the “ hp ”-approach. We thus expect in particular for n_{train} large that the (total) offline relative cost will be lower than reported in the tables.

comparable.) The online storage requirement is somewhat larger with the “ hp ”-approach, though in general the storage requirements are quite modest; the N^2 scaling moderates the growth due to K .

7 Concluding remarks

The “ hp ”-type RB method has been shown to significantly reduce the online computational cost. On the other hand, the new approach is more expensive than the standard (“ p ”-type) RB method in the offline stage. Hence we must trade offline cost for online performance. The online effort is often our main concern in the real-time or many-query contexts.

We expect the new approach to be particularly beneficial for problems for which the solution structure is very different in different parts of the parameter domain. While our model problem is specifically constructed to exhibit this property, there are many realistic problems which exhibit similar behavior. As an example, we mention an application of RB to the solution of the Fokker–Planck equation [18]; here, the solution is required for many different parameter values, but the required (“ p ”-type) RB spaces are rather large. Also of interest are problems which exhibit nonsmooth parameter dependence; the “ hp ”-approach should automatically refine the parameter domain around singularities and hence perform better than the standard approach.

There are several opportunities for extensions. First, we can generalize our approach to POD-Greedy sampling [16] for parabolic problems [9, 10]: the critical new ingredient is proper balance between additional POD modes and additional Greedy parameter values in the initial subdivision process. Second, we can extend the approach to quadratically nonlinear problems such as the incompressible Navier–Stokes equations [9]; in this case the “ hp ”-approach is particularly advantageous since the (online) computation of the error bound requires $\mathcal{O}(N^4)$ operations for N basis functions, and hence the “smaller N for larger K ” trade is particularly favorable. Third, we can consider, in the offline stage, a parallel “ hp ” approach; we can subdivide the parameter domain along each branch of the associated binary tree independently and hence concurrently.

A An improvement of Proposition 1

Let $\boldsymbol{\mu}^* \in \mathcal{D}$ be such that the RB error bound $\Delta_N(\boldsymbol{\mu}^*) = 0$; hence $\boldsymbol{\mu}^*$ corresponds to a parameter value associated with a truth snapshot residing in the RB space.

In this case the residual satisfies $r_N(v; \boldsymbol{\mu}^*) = 0$ for all $v \in X$. We now consider any $\boldsymbol{\mu} \in \mathcal{D}$. The Riesz representation of the residual at $\boldsymbol{\mu}$, $\mathcal{R}_N(\boldsymbol{\mu}) \in X$, satisfies

$$(\mathcal{R}_N(\boldsymbol{\mu}), v)_X = r_N(v; \boldsymbol{\mu}) = r_N(v; \boldsymbol{\mu}) - r_N(v; \boldsymbol{\mu}^*) \quad \forall v \in X. \quad (\text{A.1})$$

By the definition of the residual and the triangle inequality we obtain

$$\begin{aligned} |(\mathcal{R}_N(\boldsymbol{\mu}), v)_X| &\leq |f(v; \boldsymbol{\mu}) - f(v; \boldsymbol{\mu}^*)| \\ &\quad + |a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}^*) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu})| \quad \forall v \in X. \end{aligned} \quad (\text{A.2})$$

For the first term on the right-hand side of (A.2) we invoke (2.19) to obtain

$$|f(v; \boldsymbol{\mu}) - f(v; \boldsymbol{\mu}^*)| \leq \tilde{c}_2 \|v\|_X |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \quad \forall v \in X. \quad (\text{A.3})$$

For the second term on the right-hand side of (A.2) we note that

$$\begin{aligned} &|a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}^*) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu})| \\ &= |a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}^*) - a(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}) + (a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}) - a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}))| \\ &\leq |a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}^*) - a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu})| + |a(u_N(\boldsymbol{\mu}^*) - u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu})| \end{aligned} \quad (\text{A.4})$$

for all $v \in X$. (Note that the term in parentheses on the second line is equal to zero.)

For the first term on the right-hand side of (A.4) we invoke (2.18) and the Lax–Milgram lemma to obtain

$$\begin{aligned} |a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu}^*) - a(u_N(\boldsymbol{\mu}^*), v; \boldsymbol{\mu})| &\leq \tilde{c}_1 \|u_N(\boldsymbol{\mu}^*)\|_X \|v\|_X |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \\ &\leq \frac{\tilde{c}_1}{\underline{\alpha}} \|f(\cdot; \boldsymbol{\mu}^*)\|_{X'} \|v\|_X |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \\ &\leq \frac{\tilde{c}_1}{\underline{\alpha}} \max_{\boldsymbol{\mu} \in \mathcal{D}} \|f(\cdot; \boldsymbol{\mu})\|_{X'} \|v\|_X |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \end{aligned} \quad (\text{A.5})$$

for all $v \in X$. For the second term on the right-hand side of (A.4) we invoke the continuity of $a(\cdot, \cdot; \boldsymbol{\mu})$ and (2.9), and then Lemma 1, to obtain

$$\begin{aligned} |a(u_N(\boldsymbol{\mu}^*) - u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu})| &\leq \bar{\gamma} \|u_N(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu}^*)\|_X \|v\|_X \\ &\leq \tilde{C} \bar{\gamma} |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \|v\|_X \end{aligned} \quad (\text{A.6})$$

for all $v \in X$. We now combine (A.2) with (A.3), (A.4), (A.5), and (A.6) to obtain

$$(\mathcal{R}_N(\boldsymbol{\mu}), v)_X \leq |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \|v\|_X \left(\tilde{c}_2 + \max_{\boldsymbol{\mu} \in \mathcal{D}} \tilde{c}_1 \frac{\|f(\cdot; \boldsymbol{\mu})\|_{X'}}{\underline{\alpha}} + \tilde{C}\bar{\gamma} \right), \quad (\text{A.7})$$

and hence with $v = \mathcal{R}_N(\boldsymbol{\mu})$ and the expression for \tilde{C} from Lemma 1,

$$\|\mathcal{R}_N(\boldsymbol{\mu})\|_X \leq |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \tilde{C}(\underline{\alpha} + \bar{\gamma}). \quad (\text{A.8})$$

We now invoke the result (A.8) above in the context of our partition algorithm.

Assume that we have \tilde{K} subdomains, and let $\boldsymbol{\mu}^*$ correspond to the anchor point associated with the the particular subdomain that also contains $\boldsymbol{\mu}$. For the RB error bound we then get

$$\Delta_{\tilde{K}}(\boldsymbol{\mu}) \equiv \frac{\|\mathcal{R}_{\tilde{K}}(\boldsymbol{\mu})\|_X}{\alpha_{\text{LB}}(\boldsymbol{\mu})} \leq |\boldsymbol{\mu} - \boldsymbol{\mu}^*| \tilde{C} \left(1 + \frac{\bar{\gamma}}{\underline{\alpha}} \right). \quad (\text{A.9})$$

Finally, we can now replace the arguments (4.31)–(4.33) by (A.9) in order to replace the constant in Proposition 1,

$$C = \frac{2\bar{\gamma}^2 \tilde{C} |\mathcal{D}|}{\underline{\alpha}^2}, \quad (\text{A.10})$$

by the constant

$$C_{\text{im}} = 2\tilde{C} |\mathcal{D}| \left(1 + \frac{\bar{\gamma}}{\underline{\alpha}} \right). \quad (\text{A.11})$$

We note that C_{im} is an improvement over C for $\bar{\gamma}/\underline{\alpha}$ sufficiently large.

Acknowledgments

We acknowledge helpful discussions with D. Knezevic, N. C. Nguyen, D. B. P. Huynh, S. Boyaval, and B. Haasdonk during this work. We also thank D. Knezevic specifically for contributions to the alternative Proposition 1 proof given in the appendix.

Bibliography

- [1] B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16:525–528, 1978.

Paper 2

- [2] D. Amsallem, J. Cortial, and C. Farhat. On-Demand CFD-Based Aeroelastic Predictions Using a Database of Reduced-Order Bases and Models. In *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, Orlando, FL*, January 2009. AIAA Paper 2009-800.
- [3] D. Amsallem and C. Farhat. Interpolation Method for Adapting Reduced-Order Models and Application to Aeroelasticity. *AIAA Journal*, 46(7):1803–1813, July 2008.
- [4] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [5] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. Technical Report 5, Interdisciplinary Mathematics Institute, College of Arts and Sciences, University of South Carolina, 2010.
- [6] S. Boyaval. Reduced-basis approach for homogenization beyond the periodic setting. *Multiscale Modeling & Simulation*, 7(1):466–494, 2008.
- [7] S. Boyaval, C. Le Bris, Y. Maday, N. C. Nguyen, and A. T. Patera. A reduced basis approach for variational problems with stochastic parameters: Application to heat conduction with variable robin coefficient. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44):3187 – 3206, 2009.
- [8] 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. *ESAIM: Mathematical Modelling and Numerical Analysis*, submitted, November 2009. <http://augustine.mit.edu>.
- [9] J. L. Eftang, D. J. Knezevic, and A. T. Patera. An *hp* certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems*, to appear, 2011. <http://augustine.mit.edu>.
- [10] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An *hp* certified reduced basis method for parametrized parabolic partial differential equations. In J. S. Hesthaven and E. M. Rønquist, editors, *Spectral and High Order*

An hp-RB Method for Elliptic Equations

Methods for Partial Differential Equations, volume 76 of *Lecture Notes in Computational Science and Engineering*, pages 179–187. Springer Berlin Heidelberg, 2011.

- [11] J. L. Eftang and B. Stamm. Parameter multi-domain “hp” empirical interpolation. *International Journal for Numerical Methods in Engineering*, submitted, 2011. Norwegian University of Science and Technology, Numerics preprint 3/2011.
- [12] W. J. Gordon and C. A. Hall. Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, 7(4):461–477, 1973.
- [13] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [14] B. Haasdonk, M. Dihlmann, and M. Ohlberger. A Training Set and Multiple Bases Generation Approach for Parametrized Model Reduction Based on Adaptive Grids in Parameter Space. Technical Report 28, SRC SimTech, 2010.
- [15] B. Haasdonk and M. Ohlberger. Adaptive Basis Enrichment for the Reduced Basis Method Applied to Finite Volume Schemes. In *Finite Volumes for Complex Applications V, ISTE, London*, pages 471–478, 2008.
- [16] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
- [17] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique*, 345(8):473 – 478, 2007.
- [18] D. J. Knezevic and A. T. Patera. A certified reduced basis method for the fokker–planck equation of dilute polymeric fluids: Fene dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793–817, 2010.

Paper 2

- [19] N. C. Nguyen, G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes: Application to real-time bayesian parameter estimation. In Biegler, Birosa, Ghattas, Heinkenschloss, Keyes, Mallick, Marzouk, Tenorio, Waanders, and Willcox, editors, *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 151–177. John Wiley & Sons, Ltd, 2010.
- [20] N. C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous burgers’ equation. *Calcolo*, 46:157–185, 2009.
- [21] A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18:455–462, 1980.
- [22] T. A. Porsching. Estimation of the error in the reduced basis method solution of nonlinear equations. *Mathematics of Computation*, 45(172):487–496, 1985.
- [23] W. C. Rheinboldt. On the theory and error estimation of the reduced basis method for multi-parameter problems. *Nonlinear Analysis*, 21(11):849–858, 1993.
- [24] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [25] S. Sen. Reduced-basis approximation and *a posteriori* error estimation for many-parameter heat conduction problems. *Numerical Heat Transfer, Part B*, 54(5):369–389, 2008.
- [26] K. Veroy, C. Prud’homme, D. V. Rovas, and A. T. Patera. 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.

PAPER 3

**AN *hp* CERTIFIED REDUCED BASIS METHOD FOR
PARAMETRIZED PARABOLIC PARTIAL
DIFFERENTIAL EQUATIONS**

JENS L. EFTANG, DAVID J. KNEZEVIC, AND ANTHONY T. PATERA

*To appear in
Mathematical and Computer Modelling of Dynamical Systems,
2011*

AN hp CERTIFIED REDUCED BASIS METHOD FOR PARAMETRIZED PARABOLIC PARTIAL DIFFERENTIAL EQUATIONS

JENS L. EFTANG*, DAVID J. KNEZEVIC†, AND ANTHONY T. PATERA†

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

*†Department of Mechanical Engineering,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA*

Abstract

In this paper we introduce an hp certified reduced basis method for parabolic partial differential equations. We invoke a POD (in time) / Greedy (in parameter) sampling procedure first in the initial partition of the parameter domain (h -refinement) and subsequently in the construction of reduced basis approximation spaces restricted to each parameter subdomain (p -refinement). We show that proper balance between additional POD modes and additional parameter values in the initial subdivision process guarantees convergence of the approach. We present numerical results for two model problems: linear convection-diffusion, and quadratically nonlinear Boussinesq natural convection. The new procedure is significantly faster (respectively, more costly) in the reduced basis Online (respectively, Offline) stage.

1 Introduction

The certified reduced basis (RB) method is a model-order reduction framework for rapid evaluation of functional outputs, such as surface temperatures or fluxes, for partial differential equations (PDEs) which depend on an input parameter vector, for example related to geometric factors or material properties. There are four key ingredients to the certified RB framework:

- Galerkin projection: optimal linear combination of N pre-computed \mathcal{N} -degree-of-freedom “truth” finite element (FE) field snapshots [1, 17];

- POD/Greedy sampling: POD (in time) / Greedy (in parameter) [9] optimal selection and combination of FE field snapshots;
- *a posteriori* error estimation: rigorous upper bounds for the error in the RB (output) approximation with respect to the “truth” FE discretization [7, 19];
- Offline–Online computational decomposition: $\mathcal{O}(N^\bullet)$ -complexity *preprocessing* followed by $\mathcal{O}(N^\bullet)$ -complexity certified *input-output prediction* [15, 19].

We shall describe each ingredient further in subsequent sections.

We shall assume that the field variable depends smoothly on the parameters. In that case we can expect, and we can rigorously confirm *a posteriori*, that $N \ll \mathcal{N}$; we can then furthermore anticipate rapid Online evaluation of the RB output approximation and associated RB output error bound. The certified RB method is thus computationally attractive in two important engineering contexts: “real time,” such as parameter estimation and optimal control; “many query,” such as multiscale or stochastic simulation. In both instances, the Offline effort is either *unimportant* or can be *amortized* over many input-output evaluations. In both instances, rigorous error control without direct appeal to the “truth” is crucial.

For many problems, the field variable may be quite different in different regions of the parameter domain, and hence a snapshot from one region may be of little value to the RB approximation in another region. To exploit this opportunity we introduce in [5] an *hp* reduced basis method for linear elliptic equations. In the Offline stage we first adaptively subdivide the original parameter domain into smaller regions (*h*-refinement); we then construct individual RB approximation spaces spanned by snapshots restricted to parameter values within each of these parameter subdomains (*p*-refinement). In the Online stage, the RB approximation associated with any new parameter value is then constructed as a (Galerkin) linear combination of snapshots from the parameter subdomain that contains the new parameter value. The dimension of the *local* approximation space, and thus the Online cost, shall be very low: every basis function contributes significantly to the RB approximation. We note that an alternative “multiple bases generation” procedure is introduced in [8]; a different “interpolation” approach to parametric reduced order modelling with parameter subdomains is described in [2].

In this paper, we extend the work in [5] to linear and non-linear parabolic equations through a POD (in time) / Greedy (in parameter) procedure. The POD/Greedy sampling approach [9] is invoked both in the initial partition of the

parameter domain (*h*-refinement) and subsequently in the construction of RB approximation spaces restricted to each parameter subdomain (*p*-refinement). Much of the elliptic machinery from [5] extends to the parabolic case since we only subdivide the parameter (and not the temporal) domain. The critical *new* issue for the *hp*-POD/Greedy algorithm for parabolic problems is proper balance between additional POD modes and additional parameter values in the initial subdivision process.

The *hp*-POD/Greedy procedure was first introduced in the conference proceedings paper [6]. We extend [6] here in several important ways. First, we introduce an improvement to the algorithm: an additional Offline splitting step which permits direct control of the Online computational cost. Second, we introduce (for a simple but illustrative case) a new *a priori* convergence theory for the initial subdivision process; we show in particular that the procedure is convergent provided sufficiently many POD modes are included in the RB spaces. Good convergence of the subdivision process is critical both to Offline and Online performance. Third, and finally, we extend our considerations to quadratically nonlinear parabolic problems. This class of problems is particularly “ripe” for the *hp* approach due to the $\mathcal{O}(N^4)$ computational cost associated with RB error bound evaluation [12, 16]: even a small reduction in N —the number of RB basis functions—will result in significant Online computational savings.

We begin in Section 2 with the problem statement(s). In Section 3 we introduce the *hp*-RB approximation, the associated RB error bounds, and the necessary computational procedures. In Section 4 we present the *hp*-POD/Greedy algorithm and the new *a priori* convergence theory. Finally, in Section 5, we present numerical results for two model problems: a linear time-invariant (LTI) convection–diffusion problem, and a quadratically nonlinear Boussinesq natural convection problem; we focus our discussion on computational cost and Online economization compared to the standard (*p*-type) RB method.

2 Problem Statement

We directly consider a discrete-time parametrized parabolic PDE defined over a spatial domain $\Omega \subset \mathbb{R}^2$ for discrete time levels $t^k = k\Delta t$, $0 \leq k \leq K$; here $\Delta t = t_f/K$, and t_f is the final time. We further introduce a P -dimensional parameter domain $\mathcal{D} \subset \mathbb{R}^P$ and denote by $\mu \in \mathcal{D}$ a particular parameter value. For a given $\mu \in \mathcal{D}$ we shall denote the exact solution to our discrete-time parabolic PDE as $u^k(\mu) \equiv u(t^k, \mu)$, $0 \leq k \leq K$.

We consider Euler Backward ($\theta = 1$) and Crank Nicolson ($\theta = 0.5$) temporal

discretization schemes (more generally we may consider $0.5 \leq \theta \leq 1$); we define $u^{k+\theta}(\mu) \equiv \theta u^{k+1}(\mu) + (1-\theta)u^k(\mu)$. The exact formulation reads: for any $\mu \in \mathcal{D}$, find $u^k(\mu) \in X$, $1 \leq k \leq K$, such that

$$\begin{aligned} \frac{1}{\Delta t} m(u^{k+1}(\mu) - u^k(\mu), v; \mu) + a(u^{k+\theta}(\mu), v; \mu) \\ + b(u^{k+\theta}(\mu), u^{k+\theta}(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X, \end{aligned} \quad (2.1)$$

subject to initial condition $u^0(\mu)$. In the sequel we shall always assume zero initial conditions. We then evaluate our output of interest as $s^k(\mu) = \ell(u^k(\mu); \mu)$ for $0 \leq k \leq K$. Here, X denotes a Sobolev space over $\Omega \subset \mathbb{R}^2$; typically $(H_0^1(\Omega))^d \subseteq X \subseteq (H^1(\Omega))^d$, where $H^1(\Omega) = \{v : |\nabla v| \in L^2(\Omega)\}$, $H_0^1(\Omega) = \{v \in H^1(\Omega) : v|_{\partial\Omega} = 0\}$ where $\partial\Omega$ is the boundary of Ω , $L^2(\Omega)$ is the space of square integrable functions over Ω , and d is the dimension of the field. (In our exposition $d = 1$; later, for the Boussinesq problem, $d = 3$.)

We suppose that X is equipped with an inner product $(\cdot, \cdot)_X$ and induced norm $\|\cdot\|_X = (\cdot, \cdot)_X^{1/2}$; we further denote by (\cdot, \cdot) the standard $L^2(\Omega)$ inner product and by $\|\cdot\|_{L^2} = (\cdot, \cdot)^{1/2}$ the standard $L^2(\Omega)$ norm. For any $\mu \in \mathcal{D}$, $m(\cdot, \cdot; \mu)$ is a coercive and continuous bilinear form over $L^2(\Omega)$, $a(\cdot, \cdot; \mu)$ is a coercive and continuous bilinear form over X , $b(\cdot, \cdot, \cdot; \mu)$ is a continuous trilinear form over X , $f(\cdot; \mu)$ is an X -bounded linear functional, and $\ell(\cdot; \mu)$ is an $L^2(\Omega)$ -bounded linear “output” functional. We introduce coercivity constants

$$\alpha(\mu) \equiv \inf_{v \in X} \frac{a(v, v; \mu)}{\|v\|_X^2}, \quad \sigma(\mu) \equiv \inf_{v \in X} \frac{m(v, v; \mu)}{\|v\|_{L^2}^2}; \quad (2.2)$$

under our assumptions, $\alpha(\mu) > 0$ and $\sigma(\mu) > 0$ for any $\mu \in \mathcal{D}$. Note for $b = 0$ our problem is linear and coercive.

In order to develop efficient Offline-Online computational procedures for the RB field approximation, RB output approximation, and RB error bound, we shall suppose that all our forms admit “affine” expansions in functions of μ . Specifically, for any $\mu \in \mathcal{D}$

$$a(\cdot, \cdot; \mu) = \sum_{q=1}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\mu), \quad (2.3)$$

where $Q_a < Q$ and Q is finite and preferably modest. We suppose that m , b , and f admit similar expansions in at most Q terms. Many problems (including the examples of this paper) admit an affine expansion; for other problems, approximate affine representations can be developed [3, 4].

We now introduce the “truth” spatial discretization of the PDE. We suppose a regular triangulation $\mathcal{T}^{\mathcal{N}}(\Omega)$ of Ω and introduce a corresponding high-resolution finite element (FE) space $X^{\mathcal{N}} \subset X$ of dimension \mathcal{N} . The truth discretization of (2.1) reads: for any $\mu \in \mathcal{D}$, find $u^{\mathcal{N}^k}(\mu) \in X^{\mathcal{N}}$, $1 \leq k \leq K$, such that

$$\begin{aligned} \frac{1}{\Delta t} m(u^{\mathcal{N}^{k+1}}(\mu) - u^{\mathcal{N}^k}(\mu), v; \mu) + a(u^{\mathcal{N}^{k+\theta}}(\mu), v; \mu) \\ + b(u^{\mathcal{N}^{k+\theta}}(\mu), u^{\mathcal{N}^{k+\theta}}(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X^{\mathcal{N}}, \end{aligned} \quad (2.4)$$

subject to initial condition $u^{\mathcal{N}^0} = 0$; then evaluate the truth output approximation as $s^{\mathcal{N}^k}(\mu) = \ell(u^{\mathcal{N}^k}(\mu); \mu)$ for $0 \leq k \leq K$. It is this truth FE approximation that we wish to accelerate by RB treatment. We shall assume that $X^{\mathcal{N}}$ is rich enough that the exact and truth solutions are indistinguishable at the desired level of numerical accuracy. As we shall observe below, the RB Online computational cost is independent of \mathcal{N} , and the RB approximation is stable as $\mathcal{N} \rightarrow \infty$. We can thus choose \mathcal{N} conservatively.

3 hp Reduced Basis Approximation

For a parameter domain $\mathcal{D} \subset \mathbb{R}^P$, the hp -RB method serves to construct a hierarchical partition of \mathcal{D} into M distinct parameter subdomains $\mathcal{V}_{B^m} \subset \mathcal{D}$, $1 \leq m \leq M$. Each of these subdomains \mathcal{V}_{B^m} has associated nested RB approximation spaces $X_{1,B^m} \subset \cdots \subset X_{N_{\max}, B^m}$, where $\dim(X_{N, B^m}) = N$, $1 \leq N \leq N_{\max, B^m}$. We define $N_{\max} \equiv \max_{1 \leq m \leq M} N_{\max, B^m}$. The procedure for the construction of the parameter domain partition and associated RB spaces, as well as the form of the “identifiers” B^m , shall be made explicit in Section 4. In this Section, we discuss the RB approximation, the RB *a posteriori* error estimators, and the associated computational procedures *given* the parameter domain partition and associated RB spaces.

3.1 Reduced Basis Approximation

For any new $\mu \in \mathcal{D}$ we first determine $m^* \in [1, M]$ such that $\mu \in \mathcal{V}_{B^{m^*}} (\subset \mathcal{D})$. Given any N , we define $\hat{N} \equiv \min\{N, N_{\max, B^{m^*}}\}$. The RB approximation of

(2.4) reads: for any $\mu \in \mathcal{D}$, find $u_N^k(\mu) \in X_N \equiv X_{\hat{N}, B^{m^*}}$, $1 \leq k \leq K$, such that

$$\begin{aligned} \frac{1}{\Delta t} m(u_N^{k+1}(\mu) - u_N^k(\mu), v; \mu) + a(u_N^{k+\theta}(\mu), v; \mu) \\ + b(u_N^{k+\theta}(\mu), u_N^{k+\theta}(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_N, \end{aligned} \quad (3.1)$$

subject to initial condition $u_N^0 = 0$; then evaluate the RB output approximation as $s_N^k(\mu) = \ell(u_N^k(\mu); \mu)$ for $0 \leq k \leq K$.

3.2 *A posteriori* error estimation

A rigorous *a posteriori* upper bound for the RB error is crucial for the Offline *hp*-POD/Greedy sampling procedure as well as for the Online certification of the RB approximation and RB output. The key computational ingredients of the RB error bound are the RB residual dual norm and lower bounds for the stability constants.

Given an RB approximation, $u_N^k(\mu)$, $0 \leq k \leq K$, for $\mu \in \mathcal{D}$, we write the RB residual, $r_N^k(v; \mu)$, $1 \leq k \leq K$, as

$$\begin{aligned} r_N^{k+1}(v; \mu) = f(v; \mu) - \frac{1}{\Delta t} m(u_N^{k+1}(\mu) - u_N^k(\mu), v; \mu) \\ - a(u_N^{k+\theta}(\mu), v; \mu) - b(u_N^{k+\theta}(\mu), u_N^{k+\theta}(\mu), v; \mu), \quad \forall v \in X^{\mathcal{N}}. \end{aligned} \quad (3.2)$$

The Riesz representation of the residual $\hat{e}_N^k(\mu) \in X^{\mathcal{N}}$, $1 \leq k \leq K$, satisfies

$$(\hat{e}_N^k(\mu), v)_X = r_N^k(v; \mu), \quad \forall v \in X^{\mathcal{N}}. \quad (3.3)$$

We denote by $\varepsilon_N^k(\mu) = \|\hat{e}_N^k(\mu)\|_X = \sup_{v \in X^{\mathcal{N}}} \frac{r_N^k(v; \mu)}{\|v\|_X}$ the residual dual norm.

We next introduce positive lower bounds for the coercivity constants of m and a , σ_{LB} and α_{LB} , respectively, such that for all $\mu \in \mathcal{D}$

$$0 < \sigma_{\text{LB}}(\mu) \leq \sigma(\mu), \quad 0 < \alpha_{\text{LB}}(\mu) \leq \alpha(\mu). \quad (3.4)$$

We also introduce a lower bound for the (possibly negative) stability constant

$$\rho_N(t^{k+1}; \mu) \equiv \inf_{v \in X^{\mathcal{N}}} \frac{2b(u_N^{k+\theta}(\mu), v, v; \mu) + a(v, v; \mu)}{\|v\|_{L_2}^2}, \quad 0 \leq k \leq K-1, \quad (3.5)$$

which we shall denote $\rho_N^{\text{LB}}(t^k; \mu)$: $\rho_N^{\text{LB}}(t^k; \mu) \leq \rho_N(t^k; \mu)$ for $1 \leq k \leq K$ and all $\mu \in \mathcal{D}$. We further define $\tau_N^{\text{LB}}(t^k; \mu) = \min(\rho_N^{\text{LB}}(t^k; \mu), 0)$.

We can then develop the $L^2(\Omega)$ error bound

$$\Delta_N^k(\mu) = \sqrt{\frac{\Delta t \sum_{k'=1}^k \left(\frac{\varepsilon_N(t^{k'}; \mu)^2}{1 - (1-\theta)\Delta t \tau_N^{\text{LB}}(t^{k'}; \mu)} \prod_{j=1}^{k'-1} \frac{1 + \theta \Delta t \tau_N^{\text{LB}}(t^j; \mu)}{1 - (1-\theta)\Delta t \tau_N^{\text{LB}}(t^j; \mu)} \right)}{\alpha_{\text{LB}}(\mu) \sigma_{\text{LB}}(\mu) \prod_{k'=1}^k \frac{1 + \theta \Delta t \tau_N^{\text{LB}}(t^{k'}; \mu)}{1 - (1-\theta)\Delta t \tau_N^{\text{LB}}(t^{k'}; \mu)}}} \quad (3.6)$$

for which it can be demonstrated [7, 12, 16] that $\|u^{\mathcal{N}^k}(\mu) - u_N^k(\mu)\|_{L^2} \leq \Delta_N^k(\mu)$, $1 \leq k \leq K$, $\forall \mu \in \mathcal{D}$.¹ We can furthermore develop an RB output error bound

$$\Delta_{N,s}^k(\mu) \equiv \left(\sup_{v \in X^{\mathcal{N}}} \frac{\ell(v; \mu)}{\|v\|_{L^2}} \right) \Delta_N^k(\mu), \quad (3.7)$$

for which it can be demonstrated that $|s^{\mathcal{N}^k}(\mu) - s_N^k(\mu)| \leq \Delta_{N,s}^k(\mu)$, $1 \leq k \leq K$, $\forall \mu \in \mathcal{D}$.

3.3 Computational Procedures

Construction-Evaluation. Thanks to the ‘‘affine’’ assumption (2.3) we can develop Construction-Evaluation procedures for the RB field, RB output, and RB error bound. We first consider the RB field and RB output. In the *Construction* stage, given the RB basis functions, we form and store all the necessary parameter-independent entities at cost $\mathcal{O}(\mathcal{N}^\bullet)$. In the *Evaluation* stage, we first determine the subdomain to which the given new parameter μ belongs: an $\mathcal{O}(\log_2 M)$ binary search suffices thanks to the hierarchical subdomain construction which we will make explicit in the next section [5]. We next assemble the RB system (3.1) at cost $\mathcal{O}(QN^2)$ ($N \leq N_{\max}$) in the LTI case [15] and at cost $\mathcal{O}(n_{\text{Newton}}QN^3K)$ in the quadratically nonlinear case [12, 16]; we then solve this system at cost $\mathcal{O}(N^3 + KN^2)$ in the LTI case and at cost $\mathcal{O}(n_{\text{Newton}}KN^3)$ in the quadratically nonlinear case. (Here n_{Newton} is the number of Newton iterations required to solve the nonlinear equations at each timestep.) Given the RB field, the RB output can be evaluated at cost $\mathcal{O}(KN)$.

We next consider the RB error bound (3.6). We invoke the Riesz representation of the residual and linear superposition in order to develop Construction-Evaluation procedures for the residual dual norm.² In the *Construction* stage,

¹In the linear case $b = 0$, and it thus follows from (3.5) and the definition of τ_N^{LB} (we recall that $a(\cdot, \cdot; \mu)$ is coercive) that (3.6) simplifies to $\Delta_N^k(\mu) = \left(\frac{\Delta t}{\alpha_{\text{LB}} \sigma_{\text{LB}}(\mu)} \sum_{k'=1}^k \varepsilon_N(t^{k'}; \mu)^2 \right)^{1/2}$.

²We refer to [10, 16] for details on the Construction-Evaluation procedure for the computation of lower bounds for the stability constants—a Successive Constraint Method (SCM).

we again compute and store all the necessary parameter-independent entities at cost $\mathcal{O}(\mathcal{N}^\bullet)$. In the *Evaluation* stage, we can evaluate the residual dual norm at cost $\mathcal{O}(KN^2 + Q^2N^2)$ for LTI problems [15] and at cost $\mathcal{O}(KQ^2N^4)$ for quadratically nonlinear problems [12, 16]. (In the sequel we shall assume $Q = \mathcal{O}(1)$, as is the case in our numerical examples.) We note that the $\mathcal{O}(N^4)$ cost for quadratically nonlinear problems compromises rapid evaluation for larger N and in practice limits N_{\max} —motivation for an *hp* approach.

Offline-Online Decomposition: The Construction-Evaluation procedures enable efficient Offline-Online decomposition for the computation of the RB field approximation, RB output approximation, and RB output error bound. The Offline stage, which is performed only once as preprocessing, can be very expensive— \mathcal{N} -dependent complexity; the Online stage, which is typically performed many times, is comparably inexpensive— \mathcal{N} -independent complexity. We note that our RB formulation (3.1) inherits the temporal discretization of the truth (2.4); we may thus *not* choose Δt arbitrarily small without compromise to RB Online cost.

In the *hp*-RB *Offline* stage we perform the *hp*-POD/Greedy sampling procedure which we discuss in the next section and which is the focus of this paper: we invoke Construction-Evaluation procedures to identify good RB spaces and to compute and store the Construction quantities required in the Online stage. The link between the Offline and Online stages is the permanent storage of the *Online Dataset*; the storage requirement for the *hp*-RB method is $\mathcal{O}(MN_{\max}^2)$ in the linear case and $\mathcal{O}(MN_{\max}^4)$ in the quadratically nonlinear case. We recall that M is the number of subdomains identified by the *hp*-POD/Greedy. In the *hp*-RB *Online* stage we perform Evaluation based on the Online Dataset: we calculate the RB field approximation, the RB output approximation, and the RB error bound at the given new parameter in $\mathcal{O}(N^\bullet)$ complexity.

4 *hp*-POD/Greedy Sampling

In this section, we discuss the *hp*-POD/Greedy procedure for the construction of the parameter subdomain partition and the associated RB approximation spaces. We employ a hierarchical parameter domain splitting procedure and hence we may organize the subdomains in a binary tree. Let L denote the number of levels in the tree. For $1 \leq l \leq L$, we introduce Boolean vectors

$$B_l = (1, i_1, i_2, \dots, i_l) \in \{1\} \times \{0, 1\}^l. \quad (4.1)$$

For any B_l , $1 \leq l \leq L - 1$ we define the concatenation $(B_l, i) \equiv (1, i_1, \dots, i_l, i)$, $i \in \{0, 1\}$. The M subdomains of \mathcal{D} are associated to the M leaf nodes of the binary tree; we denote by B^m , $1 \leq m \leq M$, the Boolean vectors that correspond to the leaf nodes; we can thus label the parameter subdomains as $\mathcal{V}_{B^m} \subset \mathcal{D}$, $1 \leq m \leq M$. Similarly, we denote by $X_{1, B^m} \subset \dots \subset X_{N_{\max}, B^m, B^m} (\subset X^{\mathcal{N}})$ the set of nested RB approximation spaces associated to \mathcal{V}_{B^m} , $1 \leq m \leq M$.

4.1 Procedure

The *hp*-POD/Greedy algorithm introduced here applies to both the linear and non-linear case. However, we adopt the notation of the linear ($b = 0$) and scalar ($d = 1$) problem for simplicity.

Algorithm 1 $\{[\chi^i \in X, 1 \leq i \leq \Delta N]\} = \text{POD}(\{w^k \in X^{\mathcal{N}}, 1 \leq k \leq K\}, \Delta N)$

- 1: $C_{ij} \leftarrow (w^i, w^j)_X / K$, $1 \leq i, j \leq K$;
 - 2: Solve $C\psi^i = \lambda^i \psi^i$, $(\psi^i)^T C \psi^i = \frac{1}{K}$, for $(\psi^i \in \mathbb{R}^K, \lambda^i \in \mathbb{R})$ associated with the ΔN largest eigenvalues of C ;
 - 3: Compute $\chi^i = \sum_{k=1}^K \psi_k^i w^k$ for $1 \leq i \leq \Delta N$.
-

We introduce as Algorithm 1 the POD algorithm (the Method of Snapshots [20]). For specified ΔN and $\{w^k \in X^{\mathcal{N}}, 1 \leq k \leq K\}$, Algorithm 1 returns $\Delta N \leq K$ X -orthonormal functions³ $\{\chi^i \in X, 1 \leq i \leq \Delta N\}$ such that $\mathcal{P}_{\Delta N} = \text{span}\{\chi^i, 1 \leq i \leq \Delta N\}$ satisfies the optimality property

$$\mathcal{P}_{\Delta N} = \arg \inf_{\substack{Y \subset \text{span}\{w^k, 1 \leq k \leq K\} \\ \dim Y \leq \Delta N}} \left(\frac{1}{K} \sum_{k=1}^K \inf_{w \in Y} \|w^k - w\|_X^2 \right)^{1/2}. \quad (4.2)$$

The set $\{\chi^i, 1 \leq i \leq \Delta N\}$ returned by Algorithm 1 contains the ΔN first *POD modes* of $\text{span}\{w^1, \dots, w^K\}$.

We next introduce as Algorithm 2 the POD/Greedy sampling procedure of [9] (see also [13]). Let $\mathcal{V} \subseteq \mathcal{D}$. For specified ΔN , an RB space dimension upper bound \bar{N} , an initial parameter value $\mu^* \in \mathcal{V}$, a finite train sample $\Xi_{\text{train}} \subset \mathcal{V}$, and an error bound tolerance ϵ , Algorithm 2 returns $\tilde{N}_{\max} \leq \bar{N}$ nested RB spaces $X_1 \subset \dots \subset X_{\tilde{N}_{\max}}$ (note that since the spaces are nested by construction we only specify $X_{\tilde{N}_{\max}}$ as the return argument) and $\epsilon_{\max} = \max_{\mu \in \Xi_{\text{train}}} \Delta_{\tilde{N}_{\max}}^K(\mu)$

³We note that $(\chi^i, \chi^j)_X = \sum_{k=1}^K \sum_{l=1}^K \psi_k^i \psi_l^j (w^k, w^l)_X = K(\psi^i)^T C \psi^j = \delta_{ij}$.

Algorithm 2 $[X_{\tilde{N}_{\max}}, \epsilon_{\max}] = \text{POD/Greedy}(\Delta N, \bar{N}, \epsilon, \mu^*, \Xi_{\text{train}})$

- 1: Set $X_N = \{0\}$, $N = 0$, $\epsilon_{\max} = \infty$;
 - 2: **while** $\epsilon_{\max} > \epsilon$ and $N < \bar{N}$ **do**
 - 3: $e_{N,\text{proj}}^k(\mu^*) \leftarrow u^{\mathcal{N}^k}(\mu^*) - \text{proj}_{X_N}(u^{\mathcal{N}^k}(\mu^*))$, $1 \leq k \leq K$;
 - 4: **for** $i = 1, \dots, \min\{\Delta N, \bar{N} - N\}$ **do**
 - 5: $X_{N+i} \leftarrow X_N \oplus \text{span}\{\text{POD}(\{e_{N,\text{proj}}^k(\mu^*), 1 \leq k \leq K\}, i)\}$;
 - 6: **end for**
 - 7: $\mu^* \leftarrow \arg \max_{\mu \in \Xi_{\text{train}}} \Delta_N^K(\mu)$;
 - 8: $\epsilon_{\max} \leftarrow \Delta_N^K(\mu^*)$;
 - 9: $N \leftarrow N + \Delta N$;
 - 10: **end while**
 - 11: $\tilde{N}_{\max} \leftarrow N$;
-

such that either $\epsilon_{\max} \leq \epsilon$ or $\tilde{N}_{\max} = \bar{N}$. (Note in the POD/Greedy we may take the $L^2([0, t_f]; X)$ RB error bound $\Delta_{N,X}^K$ rather than the $L^2(\Omega)$ RB error bound Δ_N^K [13]; for the linear coercive case, $\Delta_{N,X}^K(\mu) = \sigma_{\text{LB}}^{1/2}(\mu) \Delta_N^K(\mu)$.)

We initialize the POD/Greedy by setting $N = 0$, $X_N = \{0\}$, and $\epsilon_{\max} = \infty$. Then, while the dimension of the RB space is less than \bar{N} and the tolerance ϵ is not satisfied over Ξ_{train} , we enrich the RB space: we first compute the projection error $e_{N,\text{proj}}^k(\mu^*) = u^{\mathcal{N}^k}(\mu^*) - \text{proj}_{X_N}(u^{\mathcal{N}^k}(\mu^*))$, $1 \leq k \leq K$, where $\text{proj}_{X_N}(w)$ denotes the X -orthogonal projection of $w \in X^{\mathcal{N}}$ onto X_N ; we next increase the dimension of the RB space by adding the ΔN first POD modes of the projection error to the current RB space; we then greedily determine the next parameter value over Ξ_{train} based on the *a posteriori* error estimator at the final time. We invoke Construction-Evaluation procedures for the computation of the maximum RB error bound over Ξ_{train} (line 7 of Algorithm 2); since the RB error bound calculation is very fast (\mathcal{N} -independent in the limit of many evaluations), we may choose Ξ_{train} very dense.

We finally introduce as Algorithm 3 the *hp*-POD/Greedy algorithm. For specified ΔN , an RB space dimension upper bound \bar{N} , error bound tolerances ϵ_{tol}^1 and ϵ_{tol}^2 , an initial parameter *anchor point* $\hat{\mu}_{(1)}^0$, and an initial train sample $\Xi_{\text{train},(1)} \subset \mathcal{D}$ of cardinality n_{train} , Algorithm 3 constructs a hierarchical splitting of \mathcal{D} into $M = M(\epsilon_{\text{tol}}^1, \bar{N})$ subdomains \mathcal{V}_{B^m} , $1 \leq m \leq M$, and associates to each parameter subdomain an RB space X_{N_{\max}, B^m, B^m} of dimension $N_{\max, B^m} \leq N_{\max} \leq \bar{N}$ such that for each subdomain \mathcal{V}_{B^m} the tolerance $\epsilon_{\text{tol}}^1 > 0$ is satisfied over $\Xi_{\text{train}, B^m} \subset \mathcal{V}_{B^m}$ by $\tilde{\Delta}_{R, B^m}^K$ and the tolerance ϵ_{tol}^2 is satisfied over Ξ_{train, B^m}

Algorithm 3 *hp*-POD/Greedy($\Xi_{\text{train},B_l}, \hat{\mu}_{B_l}^0, \epsilon_{\text{tol}}^1, \epsilon_{\text{tol}}^2, \bar{N}, \Delta N$)

- 1: Set $R \leftarrow 0, \tilde{X}_{R,B_l} \leftarrow \{0\}$;
- 2: Compute $u^{N^k}(\hat{\mu}_{B_l}^0), 1 \leq k \leq K$;
- 3: **while** $\tilde{\Delta}_{R,B_l}^K(\hat{\mu}_{B_l}^0) > \epsilon_{\text{tol}}^1/\eta$ **do**
- 4: $R \leftarrow R + 1$;
- 5: $\tilde{X}_{R,B_l} \leftarrow \text{span}\{\text{POD}(\{u^{N^k}(\hat{\mu}_{B_l}^0), 1 \leq k \leq K\}, R)\}$;
- 6: **end while**
- 7: $\hat{\mu}_{B_l}^1 \leftarrow \arg \max_{\mu \in \Xi_{\text{train},B_l}} \tilde{\Delta}_{R,B_l}^K(\mu)$ and set $\hat{\mu}_{(B_l,0)}^0 \leftarrow \hat{\mu}_{B_l}^0, \hat{\mu}_{(B_l,1)}^0 \leftarrow \hat{\mu}_{B_l}^1$;
- 8: **if** $\max_{\mu \in \Xi_{\text{train},B_l}} \tilde{\Delta}_{R,B_l}^K(\mu) > \epsilon_{\text{tol}}^1$ **then**
- 9: Determine $\Xi_{\text{train},(B_l,0)}, \Xi_{\text{train},(B_l,1)}$;
- 10: $X_{N_{\max},(B_l,0),(B_l,0)} \leftarrow \text{hp-POD/Greedy}(\Xi_{\text{train},(B_l,0)}, \hat{\mu}_{(B_l,0)}^0, \epsilon_{\text{tol}}^1, \epsilon_{\text{tol}}^2, \bar{N}, \Delta N)$;
- 11: $X_{N_{\max},(B_l,1),(B_l,1)} \leftarrow \text{hp-POD/Greedy}(\Xi_{\text{train},(B_l,1)}, \hat{\mu}_{(B_l,1)}^0, \epsilon_{\text{tol}}^1, \epsilon_{\text{tol}}^2, \bar{N}, \Delta N)$;
- 12: **else**
- 13: $[X_{N_{\max},B_l,B_l}, \epsilon_{\max}] = \text{POD/Greedy}(\Delta N, \bar{N}, \epsilon_{\text{tol}}^2, \hat{\mu}_{B_l}^0, \Xi_{\text{train},B_l})$;
- 14: **if** $\epsilon_{\max} > \epsilon_{\text{tol}}^2$ **then**
- 15: Discard X_{N_{\max},B_l,B_l} ;
- 16: Determine $\Xi_{\text{train},(B_l,0)}, \Xi_{\text{train},(B_l,1)}$;
- 17: $X_{N_{\max},(B_l,0),(B_l,0)} \leftarrow \text{hp-POD/Greedy}(\Xi_{\text{train},(B_l,0)}, \hat{\mu}_{(B_l,0)}^0, \epsilon_{\text{tol}}^1, \epsilon_{\text{tol}}^2, \bar{N}, \Delta N)$;
- 18: $X_{N_{\max},(B_l,1),(B_l,1)} \leftarrow \text{hp-POD/Greedy}(\Xi_{\text{train},(B_l,1)}, \hat{\mu}_{(B_l,1)}^0, \epsilon_{\text{tol}}^1, \epsilon_{\text{tol}}^2, \bar{N}, \Delta N)$;
- 19: **else**
- 20: Let $m = (\text{number of spaces returned so far} + 1)$ and set $B^m \equiv B_l$;
- 21: **return** $X_{N_{\max},B^m,B^m} \equiv X_{N_{\max},B_l,B_l}$;
- 22: **end if**
- 23: **end if**

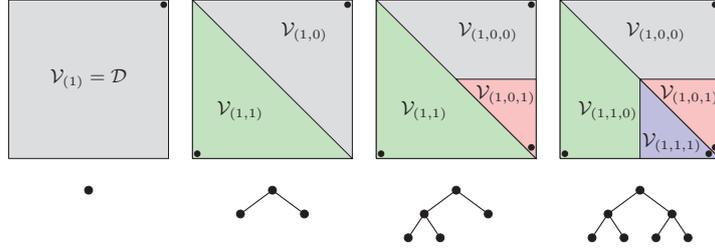


Figure 1: Two levels of h -refinement and associated binary tree; here $L = 3$.

by Δ_{N_{\max}, B^m}^K . We introduce here $\tilde{\Delta}_{R, B_l}^K$ as the RB error bound associated with the *temporary* space \tilde{X}_{R, B_l} , and we recall that Δ_{N_{\max}, B^m}^K is the RB error bound associated with the *returned* space X_{N_{\max}, B^m, B^m} . (In the hp -RB Online stage we may readily extract spaces $X_{N, B^m} \subset X_{N_{\max}, B^m}$ of any dimension N , $1 \leq N \leq N_{\max}, B^m$.)

We now comment on the constant $\eta > 1$, which in turn determines the dimension R of the temporary spaces \tilde{X}_{R, B_l} (lines 3-6): we successively increment R and evaluate $\tilde{\Delta}_{R, B_l}^K(\hat{\mu}_{B_l}^0)$ until $\tilde{\Delta}_{R, B_l}^K(\hat{\mu}_{B_l}^0) < \epsilon_{\text{tol}}^1/\eta$. For $\eta > 1$, the tolerance ϵ_{tol}^1 is then satisfied by $\tilde{\Delta}_{R, B_l}^K$ in a neighborhood of the anchor point $\hat{\mu}_{B_l}^0$, and we thus avoid arbitrarily small subdomains. We note that $\eta = \infty$ corresponds to $R = K$; however, typically $R \ll K$ is sufficient and we may thus choose η close to (but larger than) unity.

We next consider the splitting of any particular subdomain $\mathcal{V}_{B_l} \subset \mathcal{D}$ into two new subdomains $\mathcal{V}_{(B_l, 0)} \subset \mathcal{V}_{B_l}$ and $\mathcal{V}_{(B_l, 1)} \subset \mathcal{V}_{B_l}$. We suppose that \mathcal{V}_{B_l} is equipped with a train sample $\Xi_{\text{train}, B_l} \subset \mathcal{V}_{B_l}$. Given a parameter anchor point $\hat{\mu}_{B_l}^0 \in \mathcal{V}_{B_l}$, we first compute the truth field $u^{\mathcal{N}^k}(\hat{\mu}_{B_l}^0)$, $1 \leq k \leq K$, and define the temporary RB space \tilde{X}_{R, B_l} associated with the subdomain \mathcal{V}_{B_l} as discussed above. The next step is to evaluate $\tilde{\Delta}_{R, B_l}^K(\mu)$ for all $\mu \in \Xi_{\text{train}, B_l}$ in order to identify a second anchor point (line 7) $\hat{\mu}_{B_l}^1 = \arg \max_{\mu \in \Xi_{\text{train}, B_l}} \tilde{\Delta}_{R, B_l}^K(\mu)$. We note that the two anchor points $\hat{\mu}_{B_l}^0$ and $\hat{\mu}_{B_l}^1$ are maximally different in the sense of the RB error bound, and thus provide good initial parameter values for two new RB spaces.

We now introduce a distance function, $\delta : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$; for example we may choose Euclidean distance. We can then implicitly define two new subdomains $\mathcal{V}_{(B_l, 0)} \subset \mathcal{V}_{B_l}$ and $\mathcal{V}_{(B_l, 1)} \subset \mathcal{V}_{B_l}$ based on the distance to the two anchor

points: $\mathcal{V}_{(B_l,0)} = \{\mu \in \mathcal{V}_{B_l} : \delta(\hat{\mu}_{B_l}^0, \mu) < \delta(\hat{\mu}_{B_l}^1, \mu)\}$, and $\mathcal{V}_{(B_l,1)} = \{\mu \in \mathcal{V}_{B_l} : \delta(\hat{\mu}_{B_l}^0, \mu) \geq \delta(\hat{\mu}_{B_l}^1, \mu)\}$. Note that by this definition, parameter values that are equidistant from the two anchor points $\hat{\mu}_{B_l}^0$ and $\hat{\mu}_{B_l}^1$ belong to $\mathcal{V}_{(B_l,1)}$. The final step of splitting is to construct a new train sample associated with each of the two new subdomains (line 9). We first enrich (by adding random points, say) the current train sample $\tilde{\Xi}_{\text{train},B_l} \supset \Xi_{\text{train},B_l}$ such that $\tilde{\Xi}_{\text{train},B_l} \subset \mathcal{V}_{B_l}$ has cardinality $2n_{\text{train}}$; we then define

$$\Xi_{\text{train},(B_l,i)} \equiv \tilde{\Xi}_{\text{train},B_l} \cap \mathcal{V}_{(B_l,i)}, \quad i = 0, 1. \quad (4.3)$$

We note that we may choose the initial train sample for the *hp*-POD/Greedy to be rather sparse compared to the train sample for the standard POD/Greedy, since we effectively construct an adaptively refined train sample (over \mathcal{D}) during the parameter domain partition process. The adaptively generated *hp*-POD/Greedy (local) train sample associated with a given subdomain is typically much smaller than the (global) train sample associated with the standard POD/Greedy.

We apply this splitting scheme recursively in order to partition \mathcal{D} into the final M subdomains; we can thus organize the subdomains in a binary tree. In Figure 1 we illustrate the procedure, as well as the associated binary tree, for two levels of recursive splitting.

The final step is *p*-refinement: we identify the nested RB spaces to be associated with the subdomain (line 13). If the POD/Greedy returns with $\epsilon_{\text{max}} > \epsilon_{\text{tol}}^2$, we discard the generated basis and successively perform additional subdomain splitting and POD/Greedy steps until the tolerance is satisfied with at most \bar{N} basis functions (lines 15-18). This additional splitting step permits simultaneous control over ϵ_{tol}^2 and N_{max} . We note that ΔN —the number of POD modes to include at each Greedy iteration during *p*-refinement—is typically chosen small: small ΔN leads to more optimal spaces albeit at a higher (Offline) computational cost.

Under the assumption that \bar{N} is chosen such that R is always smaller than \bar{N} (note we can always “re-specify” \bar{N} if at any point $R > \bar{N}$) the *hp*-POD/Greedy algorithm provides an Online dataset such that the RB error bound tolerance ϵ_{tol}^2 is satisfied (over the train samples) with at most $N_{\text{max}} \leq \bar{N}$ basis functions.⁴ We hope to achieve this goal without the expensive execution of lines 15–18: it is our intent that if ϵ_{tol}^1 is satisfied with R basis functions, then $\epsilon_{\text{tol}}^2 < \epsilon_{\text{tol}}^1$ will be

⁴We note that when an *additional* splitting step is performed (lines 17-18), we might have to increase R beyond the requirement in line 3 in order to ensure that the anchor point $\hat{\mu}_{B_l}^1$ identified in line 7 is different from $\hat{\mu}_{B_l}^0$.

satisfied with at most $\bar{N} > R$ basis functions; whenever this is true, we discard only R basis functions at each level of splitting.

We regard lines 15–18 as insurance: if ϵ_{tol}^2 is not satisfied with at most \bar{N} basis functions—even if ϵ_{tol}^1 was satisfied with R basis functions—we discard the computed candidate space, split the subdomain, and again execute *hp*-POD/Greedy in a recursive manner. Ideally ϵ_{tol}^1 is chosen such that the insurance is rarely invoked and $N_{\max, B^m} \leq \bar{N}$ is close to \bar{N} for most $m \in [1, M]$. If the insurance is invoked too often— ϵ_{tol}^1 is too large with respect to the target \bar{N} —the Offline computational cost will be large. If the insurance is rarely or never invoked and $N_{\max, B^m} \ll \bar{N}$ for most $m \in [1, M]$, then ϵ_{tol}^1 is too small with respect to the target \bar{N} .

Remark 1. *We note that as the number of subdomains M increases, the *hp*-POD/Greedy algorithm in general requires a larger (Offline) computational cost and generates a larger Online Dataset than the standard (*p*-type) POD/Greedy method. However, in the nonlinear case, the $\mathcal{O}(N^4)$ cost and storage associated with the RB error bound helps to moderate this increase: an increase in M provides a decrease in N such that the product MN^4 grows only modestly. We further note that, thanks to the efficient $\log_2(M)$ subdomain search, M can be very large without compromise to the Online computational cost. In practice we thus seek M to balance Offline cost and Online storage against Online speed.*

Remark 2. *As discussed in [12, 16], we must employ a “nominal” lower bound ρ^* for the stability factor ρ_N for nonlinear parabolic problems during execution of the POD/Greedy: the SCM, which allows for construction of the rigorous lower bound ρ_N^{LB} , can only be performed after generation of the RB space. In this context ρ^* is a conservatively chosen constant or (say) a linear function of μ . Note that the rigor of our error bounds in the Online stage is not compromised: after completion of the POD/Greedy we perform the SCM,⁵ and subsequently the Online RB error bounds are rigorous.*

4.2 A Priori Convergence Analysis

We now introduce an *a priori* convergence theory for Algorithm 3. Selection of relatively few and optimal subdomains—small M for specified ϵ_{tol}^1 —is crucial to reduce both Offline cost and Online cost and storage. We consider here the class of linear scalar problems ($b = 0$, $d = 1$). For simplicity, we consider the case of a

⁵We note that after completion of the *hp*-POD/Greedy we can apply the SCM algorithm independently for each parameter subdomain; we thus expect a reduction in the SCM (Online) evaluation cost since the size of the parameter domain is effectively reduced.

single parameter ($P = 1$); we assume a Backward Euler temporal discretization ($\theta = 1$); and we consider the case in which $m(\cdot, \cdot; \mu)$ is parameter-independent and in particular equal to the $L^2(\Omega)$ inner product: $m(w, v; \mu) \equiv m(w, v) \equiv \int_{\Omega} wv$.

We recall that the bilinear form a and the linear functional f admit the affine expansions

$$a(\cdot, \cdot; \mu) = \sum_{q=1}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\mu), \quad f(\cdot; \mu) = \sum_{q=1}^{Q_f} f^q(\cdot) \Theta_f^q(\mu), \quad (4.4)$$

for all $\mu \in \mathcal{D}$. For our purposes in this section, we shall require that

$$a(\cdot, \cdot; \mu) = a^1(\cdot, \cdot) + \sum_{q=2}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\mu) \equiv a^1(\cdot, \cdot) + a_{\text{II}}(\cdot, \cdot; \mu), \quad (4.5)$$

where a^1 is an X -inner product and a_{II} is L^2 -continuous in its second argument. Specifically we require, for any $v \in X$, $w \in X$,

$$a^1(v, w) \leq \|v\|_X \|w\|_X, \quad (4.6)$$

$$a^q(v, w) \leq \gamma^q \|v\|_X \|w\|_{L^2}, \quad 2 \leq q \leq Q_a. \quad (4.7)$$

We also require that the $f^q : X \rightarrow \mathbb{R}$ are L^2 -bounded:

$$f^q(v) \leq \|f^q\|_{L^2} \|v\|_{L^2}, \quad 1 \leq q \leq Q_f. \quad (4.8)$$

For simplicity we suppose that $\|\cdot\|_X = \|\cdot\|_{H^1}$; hence $\|v\|_{L^2} \leq \|v\|_X$ for all $v \in X$. We further require that the $\Theta_a^q : \mathcal{D} \rightarrow \mathbb{R}$ and $\Theta_f^q : \mathcal{D} \rightarrow \mathbb{R}$ are Lipschitz continuous: for any $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$, there exists constants L_a^q and L_f^q , $1 \leq q \leq Q_a$, such that

$$|\Theta_a^q(\mu_1) - \Theta_a^q(\mu_2)| \leq L_a^q |\mu_1 - \mu_2|, \quad 1 \leq q \leq Q_a, \quad (4.9)$$

$$|\Theta_f^q(\mu_1) - \Theta_f^q(\mu_2)| \leq L_f^q |\mu_1 - \mu_2|, \quad 1 \leq q \leq Q_f. \quad (4.10)$$

We introduce lower and upper bounds over \mathcal{D} for the coercivity and continuity constants of $a(\cdot, \cdot; \mu)$:

$$0 < \underline{\alpha} \equiv \min_{\mu \in \mathcal{D}} \alpha(\mu) = \min_{\mu \in \mathcal{D}} \inf_{v \in X} \frac{a(v, v; \mu)}{\|v\|_X^2}, \quad \infty > \bar{\gamma} \geq \max_{\mu \in \mathcal{D}} \sup_{v \in X} \sup_{w \in X} \frac{a(v, w; \mu)}{\|v\|_X \|w\|_X}, \quad (4.11)$$

respectively. For simplicity of notation we suppose, for $v, w \in X$ and any $\mu \in \mathcal{D}$, that

$$a_{\Pi}(w, v; \mu) \leq \bar{\gamma} \|w\|_X \|v\|_{L^2}. \quad (4.12)$$

For our theoretical arguments below we assume $\underline{\alpha} \leq 1$ and $\bar{\gamma} \geq 1$. The coercivity lower bound $\alpha_{\text{LB}}(\mu)$ shall be given as $\alpha_{\text{LB}}(\mu) = \underline{\alpha}$ for all $\mu \in \mathcal{D}$. We emphasize that all our assumptions in this section are satisfied by our convection-diffusion numerical example of Section 5.1.

We consider Algorithm 3 with $\bar{N} = \infty$. Hence p -refinement—execution of POD/Greedy in line 13—will converge ($\epsilon_{\text{max}} \leq \epsilon_{\text{tol}}^2$) for any specified $\epsilon_{\text{tol}}^2 > 0$. We thus focus here on h -refinement; we show in particular that the hp -POD/Greedy algorithm generates a finite number of parameter subdomains.

To this end, we shall require the following continuity result.

Lemma 1. *For any $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$, and any $v \in X$, $w \in X$, there exist positive constants c_a and c_f such that*

$$|a(v, w; \mu_1) - a(v, w; \mu_2)| \leq c_a |\mu_1 - \mu_2| \|v\|_X \|w\|_{L^2}, \quad (4.13)$$

$$|f(v; \mu_1) - f(v; \mu_2)| \leq c_f |\mu_1 - \mu_2| \|v\|_{L^2}. \quad (4.14)$$

Proof. We refer to Appendix A for the proof. \square

We next define, for any $\mu \in \mathcal{D}$ and any $v^k \in X$, $1 \leq k \leq K$, the “energy-norm”

$$\|v^k\|_{\mu} \equiv \left(m(v^k, v^k) + \Delta t \sum_{k'=1}^k a(v^{k'}, v^{k'}; \mu) \right)^{1/2}. \quad (4.15)$$

We shall require the following stability result.

Lemma 2. *For any $\mu \in \mathcal{D}$, the solution $u^{\mathcal{N}^k}(\mu) \in X^{\mathcal{N}}$, $1 \leq k \leq K$, of (2.4) for $\theta = 1$ satisfies*

$$\|u^{\mathcal{N}^k}(\mu)\|_{\mu} \leq \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'} \sqrt{\frac{t^k}{\underline{\alpha}}}, \quad 1 \leq k \leq K. \quad (4.16)$$

Proof. We refer to Appendix B for the proof. \square

For $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$, and for $1 \leq k \leq K$, we define $\Delta u_{\mathcal{N}}^k \equiv u_{\mathcal{N}}^k(\mu_1) - u_{\mathcal{N}}^k(\mu_2)$. We shall require the following continuity result.

Lemma 3. Assume that $\mu_1 \in \mathcal{D}$ and $\mu_2 \in \mathcal{D}$ belong to the same parameter subdomain (say) $\mathcal{V}_{B_l} \subset \mathcal{D}$, and let X_N denote the RB space associated with \mathcal{V}_{B_l} . Let $u_N^k(\mu_1) \in X_N$ and $u_N^k(\mu_2) \in X_N$, $1 \leq k \leq K$, satisfy (3.1) for $\theta = 1$. Then

$$\|\Delta u_N^k\|_{\mu_2} \leq \tilde{C}|\mu_1 - \mu_2|, \quad 1 \leq k \leq K, \quad (4.17)$$

where

$$\tilde{C} = \left(\frac{2t^k}{\underline{\alpha}^3} \left(\underline{\alpha}^2 c_f^2 + c_a^2 \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'}^2 \right) \right)^{1/2} \quad (4.18)$$

Proof. We refer to Appendix C for the proof. \square

We shall finally require the following continuity result, which is a discrete counterpart of Proposition 11.1.11 of [18].

Lemma 4. Assume that $\mu_1 \in \mathcal{D}$ and $\mu_2 \in \mathcal{D}$ belong to the same parameter subdomain (say) $\mathcal{V}_{B_l} \subset \mathcal{D}$, and let X_N denote the RB space associated with \mathcal{V}_{B_l} . Let $u_N^k(\mu_1) \in X_N$ and $u_N^k(\mu_2) \in X_N$, $1 \leq k \leq K$, satisfy (3.1) for $\theta = 1$. Then the finite difference $(\Delta u_N^k - \Delta u_N^{k-1})/\Delta t$ is L^2 -bounded in time:

$$\left(\frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 \right)^{1/2} \leq \hat{C}|\mu_1 - \mu_2|, \quad (4.19)$$

where

$$\hat{C} = \left(\frac{3}{\underline{\alpha}^2} \left(\tilde{\gamma}^2 \underline{\alpha} \tilde{C}^2 + t^k \underline{\alpha}^2 c_f^2 + t^k c_a^2 \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'} \right) \right)^{1/2} \quad (4.20)$$

Proof. We refer to Appendix D for the proof. \square

We now claim

Proposition 1. Let $\mathcal{D} \subset \mathbb{R}$ and let $|\mathcal{D}|$ denote the length of \mathcal{D} . For specified ϵ_{tol}^1 , Algorithm 3 terminates for finite $M = M(\epsilon_{\text{tol}}^1)$ subdomains; moreover, the convergence of the h-refinement stage is first order in the sense that

$$M(\epsilon_{\text{tol}}^1) \leq \max \left\{ 1, \frac{C}{\epsilon_{\text{tol}}^1} \right\}, \quad C = C(\eta, |\mathcal{D}|). \quad (4.21)$$

Proof. The proof has two steps. We first show that the RB error bound is Lipschitz continuous. We then relate this result to our particular procedure to prove convergence of the hp -POD/Greedy algorithm.

Step 1. We recall that for $\mu \in \mathcal{D}$, the Riesz representation $\hat{e}_N^k(\mu)$ of the residual $r_N^k(\cdot; \mu)$, $1 \leq k \leq K$, satisfies

$$(\hat{e}_N^k, v)_X = r_N^k(v; \mu), \quad \forall v \in X^{\mathcal{N}}. \quad (4.22)$$

Let $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$. We define $\Delta \hat{e}_N^k \equiv \hat{e}_N^k(\mu_1) - \hat{e}_N^k(\mu_2)$. From (4.22) we note that by linearity

$$\begin{aligned} (\Delta \hat{e}_N^k, v)_X &= \underbrace{f(v; \mu_1) - f(v; \mu_2)}_{\text{I}} + \underbrace{a(u_N^k(\mu_2), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1)}_{\text{II}} \\ &\quad + \frac{1}{\Delta t} \underbrace{\left(m(u_N^k(\mu_2) - u_N^{k-1}(\mu_2), v) - m(u_N^k(\mu_1) - u_N^{k-1}(\mu_1), v) \right)}_{\text{III}}, \end{aligned} \quad (4.23)$$

for all $v \in X^{\mathcal{N}}$ and for $1 \leq k \leq K$. For the term I we invoke Lemma 1 directly to obtain

$$|f(v; \mu_1) - f(v; \mu_2)| \leq c_f |\mu_1 - \mu_2| \|v\|_X, \quad \forall v \in X. \quad (4.24)$$

For the term II we first write

$$\begin{aligned} &|a(u_N^k(\mu_2), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1)| \\ &= |a(u_N^k(\mu_1), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1) - a(\Delta u_N^k, v; \mu_2)|. \end{aligned} \quad (4.25)$$

Then, by the triangle inequality, Lemma 1, continuity, and (4.11), we obtain

$$\begin{aligned} &|a(u_N^k(\mu_2), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1)| \\ &\leq |a(\Delta u_N^k, v; \mu_2)| + c_a \|u_N^k(\mu_1)\|_X \|v\|_X |\mu_1 - \mu_2| \\ &\leq \bar{\gamma} \|\Delta u_N^k\|_X \|v\|_X + c_a \|u_N^k(\mu_1)\|_X \|v\|_X |\mu_1 - \mu_2|. \end{aligned} \quad (4.26)$$

For the term III we invoke linearity, the Cauchy-Schwarz inequality, and the Poincaré inequality⁶ to obtain

$$\begin{aligned} &|m(u_N^k(\mu_2) - u_N^{k-1}(\mu_2), v) - m(u_N^k(\mu_1) - u_N^{k-1}(\mu_1), v)| = |m(\Delta u_N^k - \Delta u_N^{k-1}, v)| \\ &\leq \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} \|v\|_{L^2} \leq \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} \|v\|_X. \end{aligned} \quad (4.27)$$

⁶We suppose here for simplicity that $\|\cdot\|_X = \|\cdot\|_{H^1}$; hence $\|v\|_{L^2} \leq \|v\|_X$ for all $v \in X$

We now insert the expressions for I, II, and III into (4.23); for $v = \Delta \hat{e}_N^k$ we then obtain

$$\begin{aligned} (\Delta \hat{e}_N^k, \Delta \hat{e}_N^k)_X &\leq c_f |\mu_1 - \mu_2| \|\Delta \hat{e}_N^k\|_X + \bar{\gamma} \|\Delta u_N^k\|_X \|\Delta \hat{e}_N^k\|_X \\ &+ c_a \|u_N^k(\mu_1)\|_X \|\Delta \hat{e}_N^k\|_X |\mu_1 - \mu_2| + \frac{1}{\Delta t} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} \|\Delta \hat{e}_N^k\|_X. \end{aligned} \quad (4.28)$$

We divide through in (4.28) by $\|\Delta \hat{e}_N^k\|_X$, square both sides, and invoke the inequality $(A + B + C + D)^2 \leq 4(A^2 + B^2 + C^2 + D^2)$ for $A, B, C, D \in \mathbb{R}$ to obtain

$$\begin{aligned} \|\Delta \hat{e}_N^k\|_X^2 &\leq 4|\mu_1 - \mu_2|^2 (c_f^2 + c_a^2 \|u_N^k(\mu_1)\|_X^2) \\ &+ \frac{4}{\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + 4\bar{\gamma}^2 \|\Delta u_N^k\|_X^2. \end{aligned} \quad (4.29)$$

We multiply through in (4.29) by Δt , substitute k for k' , and sum over k' to obtain

$$\begin{aligned} \Delta t \sum_{k'=1}^k \|\Delta \hat{e}_N^{k'}\|_X^2 &\leq 4|\mu_1 - \mu_2|^2 (c_f^2 t^k + c_a^2 \Delta t \sum_{k'=1}^k \|u_N^{k'}(\mu_1)\|_X^2) \\ &+ 4\bar{\gamma}^2 \left(\frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 + \Delta t \sum_{k'=1}^k \|\Delta u_N^{k'}\|_X^2 \right). \end{aligned} \quad (4.30)$$

Next, from coercivity and Lemma 2 we note that

$$\Delta t \sum_{k'=1}^k \|u_N^{k'}(\mu_1)\|_X^2 \leq \frac{\|u_N^k(\mu_1)\|_{\mu_1}^2}{\underline{\alpha}} \leq \frac{t^k}{\underline{\alpha}^2} \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'}^2. \quad (4.31)$$

Further, from coercivity and (4.11), and Lemma 3 and Lemma 4, we note that

$$\begin{aligned} &4\bar{\gamma}^2 \left(\frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 + \Delta t \sum_{k'=1}^k \|\Delta u_N^{k'}\|_X^2 \right) \\ &\leq 4\bar{\gamma}^2 \left(\frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 + \Delta t \sum_{k'=1}^k \frac{a(\Delta u_N^{k'}, \Delta u_N^{k'}; \mu_2)}{\underline{\alpha}} \right) \\ &\leq 4\bar{\gamma}^2 |\mu_1 - \mu_2|^2 \left(\hat{C}^2 + \frac{\tilde{C}^2}{\underline{\alpha}} \right) \end{aligned} \quad (4.32)$$

From (4.30) with (4.31) and (4.32) we thus obtain

$$\Delta t \sum_{k'=1}^k \|\Delta \hat{e}^k\|_X^2 \leq c^2 |\mu_1 - \mu_2|^2, \quad (4.33)$$

where

$$c \equiv 2 \left(\frac{t^k}{\underline{\alpha}^2} \left(\underline{\alpha}^2 c_f^2 + c_a^2 \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'}^2 \right) + \bar{\gamma}^2 \left(\tilde{C}^2 + \frac{\hat{C}^2}{\underline{\alpha}} \right) \right)^{1/2}. \quad (4.34)$$

By the definition of the RB error bound (recall that we use $\alpha_{\text{LB}}(\mu) = \underline{\alpha}$) and the reverse triangle inequality we finally obtain

$$\begin{aligned} |\Delta_N^k(\mu_1) - \Delta_N^k(\mu_2)| &\leq \left| \left(\frac{\Delta t}{\underline{\alpha}} \sum_{k'=1}^k \|\hat{e}_N^k(\mu_1)\|_X^2 \right)^{1/2} - \left(\frac{\Delta t}{\underline{\alpha}} \sum_{k'=1}^k \|\hat{e}_N^k(\mu_2)\|_X^2 \right)^{1/2} \right| \\ &\leq \left(\frac{\Delta t}{\underline{\alpha}} \sum_{k'=1}^k \|\Delta \hat{e}_N^k\|_X^2 \right)^{1/2} \leq \frac{c}{\sqrt{\underline{\alpha}}} |\mu_1 - \mu_2|. \end{aligned} \quad (4.35)$$

Step 2. The next step is to relate (4.35) to the convergence of Algorithm 3. The algorithm generates a partition of \mathcal{D} into M subdomains. Either $M = 1$, in which case the proof is complete, or $M > 1$. We now examine the case $M > 1$. We consider the splitting of any particular subdomain $\mathcal{V}_{B_i} \subset \mathcal{D}$ into two new subdomains $\mathcal{V}_{(B_i,0)} \subset \mathcal{V}_{B_i}$ and $\mathcal{V}_{(B_i,1)} \subset \mathcal{V}_{B_i}$. We denote here by $\hat{\mu}_0 = \hat{\mu}_{B_i}^0 = \hat{\mu}_{(B_i,0)}^0$ the anchor point associated with \mathcal{V}_{B_i} and $\mathcal{V}_{(B_i,0)}$, and by $\hat{\mu}_1 = \hat{\mu}_{B_i}^1 = \hat{\mu}_{(B_i,1)}^0$ the anchor point associated with $\mathcal{V}_{(B_i,1)}$. We assume that the error tolerance at the final time is not satisfied over (a train sample over) \mathcal{V}_{B_i} ; hence $\epsilon_{\text{tol}}^1 < \tilde{\Delta}_{R,B_i}^K(\hat{\mu}_1)$. We recall that by construction of our procedure $\tilde{\Delta}_{R,B_i}^K(\hat{\mu}_0) \leq \epsilon_{\text{tol}}^1/\eta$ for specified $\eta > 1$. We can thus invoke (4.35) for $\mu_1 = \hat{\mu}_1$, $\mu_2 = \hat{\mu}_0$, and Δ_N^k replaced by $\tilde{\Delta}_{R,B_i}^K$ to conclude that

$$\epsilon_{\text{tol}}^1 - \frac{\epsilon_{\text{tol}}^1}{\eta} < |\tilde{\Delta}_{R,B_i}^K(\hat{\mu}_1) - \tilde{\Delta}_{R,B_i}^K(\hat{\mu}_0)| \leq \frac{c}{\sqrt{\underline{\alpha}}} |\hat{\mu}_1 - \hat{\mu}_0|, \quad (4.36)$$

and hence

$$|\hat{\mu}_1 - \hat{\mu}_0| > \frac{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}{c\eta}. \quad (4.37)$$

We now split \mathcal{V}_{B_l} into $\mathcal{V}_{(B_l,0)}$ and $\mathcal{V}_{(B_l,1)}$ based on Euclidean distance to the two anchor points. It is clear that

$$|\mathcal{V}_{(B_l,i)}| \geq \frac{1}{2} |\hat{\mu}_1 - \hat{\mu}_0| > \frac{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}{2c\eta}, \quad i = 0, 1. \quad (4.38)$$

The partition procedure generates $M > 1$ distinct subdomains \mathcal{V}_{B^m} , $1 \leq m \leq M$.⁷ Each of these subdomains is the result of a “parent” subdomain $\mathcal{V}_{B_l} \supset \mathcal{V}_{B^m}$ (for some B_l , $1 \leq l \leq L - 1$). Since B_l above was arbitrary, we can successively set \mathcal{V}_{B_l} to be the parent of each of the M “leaf” subdomains and conclude that

$$|\mathcal{V}_{B^m}| > \frac{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}{2c\eta}, \quad 1 \leq m \leq M. \quad (4.39)$$

We define $\delta_M \equiv \min_{1 \leq m \leq M} |\mathcal{V}_{B^m}|$; hence in particular $\delta_M > \frac{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}{2c\eta}$.

We complete the proof by a contradiction argument. Assume that $M \geq \frac{|\mathcal{D}| 2c\eta}{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}$. Thus

$$M\delta_M > \frac{|\mathcal{D}| 2c\eta}{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)} \frac{\epsilon_{\text{tol}}^1 \sqrt{\underline{\alpha}} (\eta - 1)}{2c\eta} = |\mathcal{D}|, \quad (4.40)$$

which is clearly a false statement. We conclude that the number of subdomains $M = M(\epsilon_{\text{tol}}^1) < C(\eta, |\mathcal{D}|) / \epsilon_{\text{tol}}^1$ with $C(\eta, |\mathcal{D}|) = \frac{|\mathcal{D}| 2c\eta}{\sqrt{\underline{\alpha}} (\eta - 1)}$. We finally note that Algorithm 3 is convergent since the POD/Greedy (line 13) will be able to satisfy the error bound tolerance ϵ_{tol}^2 within each of the M final subdomains (recall that we set $\bar{N} = \infty$). \square

Remark 3. *The requirement $\eta > 1$ reappears in the proof in (4.36). We note that we can not obtain a positive lower bound for the distance between the two anchor points if $\eta \leq 1$.*

Remark 4. *If we assume only $f \in X'$ (and not in L^2) and furthermore a_{II} only X -continuous in both arguments (and not L^2 -continuous in the second argument), then we can still obtain Proposition 1 albeit with an additional factor $1/\Delta t$ in the “constant” C . However, we note that this $1/\Delta t$ factor is in this case relatively “benign”: we can not in any event let “ $\Delta t \rightarrow 0$ ” in practice because of*

⁷In fact, we should interpret M here as the number of subdomains generated by Algorithm 3 so far; the \mathcal{V}_{B^m} , $1 \leq m \leq M$, are not necessarily the final M subdomains. With this interpretation we thus do not presume termination of the algorithm.

the increase in Online computational cost. (In contrast, we can let “ $\mathcal{N} \rightarrow \infty$ ” since larger \mathcal{N} affects only Offline cost.)

We recall that all the hypotheses of Proposition 1 are satisfied by our numerical example in Section 5.1.

Remark 5. Proposition 1 guarantees that the h -refinement partition algorithm is convergent. However, the convergence is very slow and hence subsequent p -refinement is in practice necessary. But note that with only a global Lipschitz constant c in our proof, our bound (4.21) is very pessimistic, and in particular does not reflect any adaptivity in the partition. In practice we expect that the algorithm adaptively generates smaller subdomains in areas of \mathcal{D} for which the field exhibits larger variations with the parameters.

5 Numerical Results

We now present numerical results for two model problems. We demonstrate that in both cases the hp -RB method yields significant Online computational savings relative to a standard (p -type) RB approach; we also show that the partitions of \mathcal{D} may reflect the underlying parametric sensitivity of the problems. All our computational results are obtained via `rb00mit` [14], which is an RB plugin for the open-source FE library `libMesh` [11]. All computations are performed on a 2.66 GHz processor. For the hp -RB approximations below, we have used a “scaled” Euclidean distance for the distance function $\delta(\cdot, \cdot)$: we map \mathcal{D} (a rectangle in both our examples) to $\hat{\mathcal{D}} = [0, 1]^P$ (via an obvious affine transformation) and compute the Euclidean distance on $\hat{\mathcal{D}}$. For the constant η in Algorithm 3 we choose $\eta = 1.1$.

5.1 Convection-Diffusion Problem

We consider the nondimensional temperature u which satisfies the convection-diffusion equation in the spatial domain $\Omega = \{(x_1, x_2) : x_1^2 + x_2^2 < 2\}$ for the discrete time levels $t^k = 0.01k$, $0 \leq k \leq 100$; we employ Backward Euler temporal discretization (hence $\theta = 1$). We impose a parameter-dependent velocity field $V(\mu) \equiv V(\nu, \varphi) \equiv (\nu \cos \varphi, \nu \sin \varphi)$ and we prescribe a constant forcing term $q = 10$. We specify homogeneous Dirichlet boundary conditions and zero initial conditions. We denote a particular parameter value $\mu \in \mathcal{D}$ by $\mu = (\nu, \varphi)$ and we introduce the parameter domain $\mathcal{D} = [0, 10] \times [0, \pi] \subset \mathbb{R}^{P=2}$. For this problem, we focus for simplicity on the RB field approximation and thus we do not consider any particular outputs.

We next introduce the forms

$$\begin{aligned} m(w, v; \mu) &= \int_{\Omega} wv, \\ a(w, v; \mu) &= \int_{\Omega} \left(\nabla w \cdot \nabla v + (V(\mu) \cdot \nabla w)v \right), \\ f(v; \mu) &= q \int_{\Omega} v = 10 \int_{\Omega} v, \end{aligned} \tag{5.1}$$

for $v, w \in X$, where $X = H_0^1(\Omega)$. Our problem can then be expressed in the form (2.4) with $b = 0$; note that our only parameter-dependent form is a , which admits an affine expansion (2.3) with $Q_a = 3$. We note that this problem satisfies all the theoretical hypothesis of Proposition 1.⁸ For our truth approximation we choose a \mathbb{P}_2 FE space $X^{\mathcal{N}} \subset X$ of dimension $\mathcal{N} = 1889$.

To obtain a benchmark for comparison we first perform a standard (p -type) POD/Greedy: we specify $\epsilon = 10^{-5}$ for the target tolerance, $\Delta N = 1$ for the number of POD modes to include at each greedy iteration, $\mu^* = (0, 0)$ for the initial parameter value, and a train sample $\Xi_{\text{train}} \subset \mathcal{D}$ of size 900. We then execute Algorithm 2 (we also “specify” $\bar{N} = \infty$ such that the POD/Greedy terminates for ϵ satisfied over Ξ_{train}). The tolerance is in this case satisfied for $N_{\text{max}} = \tilde{N}_{\text{max}} = 129$.

We next perform two *hp*-POD/Greedy computations. In the first we specify $\epsilon_{\text{tol}}^1 = 5$, $\epsilon_{\text{tol}}^2 = 10^{-5}$, $\bar{N} = 65$, $\Delta N = 1$, $\hat{\mu}_{(1)}^0 = (0, 0)$, and a train sample $\Xi_{\text{train},(1)}$ of size 64. In this case Algorithm 3 terminates for $M = 22$ subdomains with $N_{\text{max}} = \bar{N} = 65$ (recall that $N_{\text{max}} \equiv \max_{1 \leq m \leq M} N_{\text{max}, B^m}$). In the second case we specify $\epsilon_{\text{tol}}^1 = 1.5$, $\epsilon_{\text{tol}}^2 = 10^{-5}$, $\bar{N} = 45$, $\Delta N = 1$, $\hat{\mu}_{(1)}^0 = (0, 0)$, and a train sample $\Xi_{\text{train},(1)}$ of size 25. In this case Algorithm 3 terminates for $M = 278$ subdomains with $N_{\text{max}} = \bar{N} = 45$. The maximum RB $L^2(\Omega)$ error bound $\epsilon_{N, M}^{\text{max}}$ (over the train samples) over all M subdomains for each of the cases $M = 22$ and $M = 278$, as well as the p -type reference case $M = 1$, are plotted in Figure 2 as functions of N . We note that larger M yields smaller N , as desired.

We show the two partitions of \mathcal{D} in Figure 3.⁹ Note that the field variable

⁸Eq. (4.5) is satisfied with $a_{\text{II}}(w, v; \mu) = \int_{\Omega} (V(\mu) \cdot \nabla w)v$. We note that a_{II} is $L^2(\Omega)$ continuous in its second argument since by the Cauchy-Schwarz inequality $a_{\text{II}}(w, v) \leq (\int_{\Omega} (V(\mu) \cdot \nabla w)^2)^{1/2} (\int_{\Omega} v^2)^{1/2}$.

⁹To ensure a good spread over \mathcal{D} of the rather few (25 or 64 for our two examples) *initial* train points, we use for $\Xi_{\text{train},(1)}$ a deterministic initial regular grid. (For the train sample *enrichment*, we use random points.) Since some train points belong to a regular grid, the procedure may produce “aligned” subdomain boundaries, as seen in Figure 3.

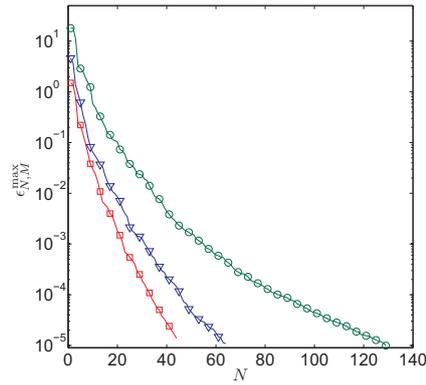


Figure 2: Convergence: hp -RB (triangles ($M = 22$) and squares ($M = 278$)) and p -type RB (circles). In the hp -RB cases, the error bound is the maximum over all subdomains for a given N .

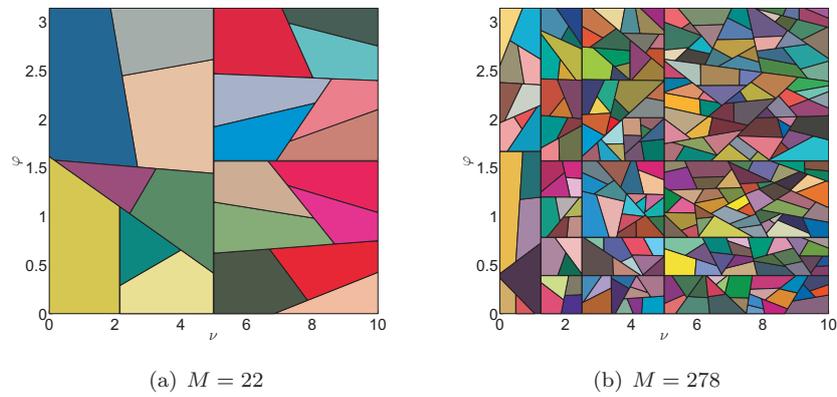


Figure 3: Parameter domain partitions \mathcal{V}_{B^m} , $1 \leq m \leq M$, for the convection-diffusion problem.

exhibits larger variations with φ for larger ν , and hence we would expect the subdomain size to decrease with increasing ν . However, this is not the case in Figure 3(b) except for smaller ν . By way of explanation we note that when the field varies significantly with *time*, which is indeed the case for large ν , R —the number of POD modes in the temporary space \tilde{X}_{R,B_i} —will be larger. We suspect that the additional POD modes included in the \tilde{X}_{R,B_i} associated with subdomains for ν larger than approximately 5 may also represent some parametric variations in the field and hence account for the “non-monotonic” (in ν) subdomain size.

We note that the hp -RB method indeed yields a significant Online speedup. Online p -type RB calculation of the RB solution coefficients and error bound for $N = 129$ basis functions requires $1.4 \cdot 10^{-2}$ seconds. In contrast, Online hp -RB calculation of the RB solution coefficients and error bound for the case with $M = 22$ subdomains and $N = 65$ requires $3.3 \cdot 10^{-3}$ seconds, and for the case with $M = 278$ subdomains and $N = 45$ requires $1.8 \cdot 10^{-3}$ seconds; in both cases, the search for the subdomain containing the new online parameter is negligible ($\mathcal{O}(10^{-6})$ seconds). (The timing results are averages over 100 Online calculations for randomly selected $\mu \in \mathcal{D}$.)

Of course Offline cost and Online storage are larger for the hp -RB than for the standard (p -type) RB: the Offline stage requires 29.6 minutes and 3.5 hours for the hp -RB computations ($M = 22$ and $M = 278$, respectively) and only 13.4 minutes for the standard RB; the Online Dataset requires 25.3 Mbytes and 142.9 Mbytes for the hp -RB computations ($M = 22$ and $M = 278$, respectively) and only 5.7 Mbytes for the standard RB. In particular Offline cost for the $M = 278$ computation is admittedly very large compared to the Offline cost for the p -type computation. Of course, even in our “real time” and “many query” contexts, the larger Offline cost associated with the hp -RB method may be an issue; we must thus seek to balance the increase in Offline cost against the decrease in Online cost by appropriate choices of the parameters ϵ_{tol}^1 and \bar{N} . We note that for this problem, our $M = 22$ hp -RB computation provides significant Online speedup at only modest increase in Offline cost.

The additional splitting step—the “insurance” provided by lines 15–18 in Algorithm 3—was never invoked for either hp -POD/Greedy computation. For the computation with specified $\bar{N} = 65$, the average of N_{\max,B^m} , $1 \leq m \leq M = 22$ is 57.3. For the computation with specified $\bar{N} = 45$, the average of N_{\max,B^m} , $1 \leq m \leq M = 278$, is 37.9. We conclude that in both cases we could have chosen ϵ_{tol}^1 somewhat larger (at the risk of invoking insurance) in order to obtain a more optimal partition with respect to the target \bar{N} .

We finally note that calculation of the truth (2.4) for this problem with $\mathcal{N} = 1889$ requires about 0.9 seconds. The average speedup relative to a truth calculation is approximately 64 for the p -type Online calculation with $N = 129$, and approximately 273 and 500 for the hp -RB Online calculations ($N = 65$, $M = 22$, and $N = 45$, $M = 278$, respectively).

5.2 Boussinesq Problem

We consider natural convection in the two-dimensional enclosure $\Omega = (0, 5)^2 \setminus \mathcal{P}$, where \mathcal{P} is the “pillar” $(2.5 - 0.1, 2.5 + 0.1) \times (0, 1)$, for the discrete time levels $t^k = 0.0016k$, $0 \leq k \leq 100$; we employ Crank-Nicolson temporal discretization (hence $\theta = 0.5$). The direction of the acceleration of gravity is defined by the unit vector $(-\sin \phi, -\cos \phi)$. We solve for the field variables V_1, V_2 (the x and y components of the fluid velocity) and ϑ (the temperature) over Ω ; hence the field has dimension $d = 3$. The “roof” of the enclosure is maintained at temperature $\vartheta = 0$, the sides and base of the enclosure are perfectly thermally insulated, and the top and sides of the pillar are subject to a uniform heat flux of magnitude Gr (the Grashof number); we impose no-slip velocity conditions on all walls. We denote a particular parameter value $\mu \in \mathcal{D}$ by $\mu = (\mu_1, \mu_2) = (\text{Gr}, \phi)$ and we introduce the parameter domain $\mathcal{D} = [4000, 6000] \times [0, 0.2] \subset \mathbb{R}^{P=2}$. Note we set the Prandtl number, Pr , here to 0.71 (for air).

Our goal is to study parametric dependence of the temperature in regions at or near the top of the heated pillar (or “fin”) in the presence of natural convection, and hence we are interested in local average-temperature outputs. These outputs can be expressed as $L^2(\Omega)$ -bounded functionals of ϑ , namely,

$$s_n(t; \mu) = \ell_n(\vartheta(t; \mu), \mu) = \frac{1}{\mu_1 |D_n|} \int_{D_n} \vartheta(t; \mu); \quad (5.2)$$

here $D_1 = [2.2, 2.4] \times [1, 1.1]$, $D_2 = [2.4, 2.6] \times [1, 1.1]$, $D_3 = [2.6, 2.8] \times [1, 1.1]$ are three small rectangles above the pillar. The domain geometry and output regions are depicted in Figure 4.

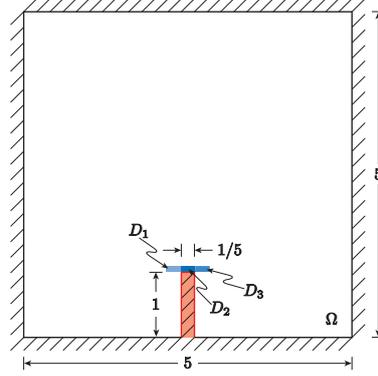


Figure 4: The computational domain; note that Ω does not include the pillar, which is shaded in red. The output regions D_1, D_2 and D_3 are also indicated.

We introduce the forms

$$\begin{aligned}
 m(w, v; \mu) &= \int_{\Omega} w_i v_i, \\
 a(w, v; \mu) &= \int_{\Omega} \left(\frac{\partial w_1}{\partial x_j} \frac{\partial v_1}{\partial x_j} + \frac{\partial w_2}{\partial x_j} \frac{\partial v_2}{\partial x_j} + \frac{1}{\text{Pr}} \frac{\partial w_3}{\partial x_j} \frac{\partial v_3}{\partial x_j} \right), \\
 b_1(w, v; \mu) &= -\sqrt{\mu_1 \text{Pr}} \sin \mu_2 \int_{\Omega} w_3 v_1 - \sqrt{\mu_1 \text{Pr}} \cos \mu_2 \int_{\Omega} w_3 v_2, \quad (5.3) \\
 b_2(w, z, v; \mu) &= \frac{1}{2\sqrt{\mu_1 \text{Pr}}} \int_{\Omega} \left(\frac{\partial w_i z_j}{\partial x_j} + z_j \frac{\partial w_i}{\partial x_j} \right) v_i, \\
 f(v; \mu) &= \frac{\mu_1}{\text{Pr}} \int_{\partial\Omega_p} v_3,
 \end{aligned}$$

for $w = (w_1, w_2, w_3) \in X$, $v = (v_1, v_2, v_3) \in X$, and $z = (z_1, z_2, z_3) \in X$; in these expressions, $i = 1, 2, 3$ and $j = 1, 2$. Here, $X = Z \times W$, where Z is the divergence-free subspace of $(H_0^1(\Omega))^2$, and $H_0^1(\Omega) \subset W \subset H^1(\Omega)$ is the subspace of $H^1(\Omega)$ of functions which vanish on the enclosure roof.

Our problem can then be expressed in the form (2.4) with $b(w, z, v; \mu) = b_1(w, v; \mu) + b_2(w, z, v; \mu)$ (we have used a skew-symmetric form of the nonlinear convection operator $b_2(w, z, v; \mu)$ in order to generate certain discrete stability properties [18]); note that all forms satisfy the ‘‘affine’’ assumption. For our

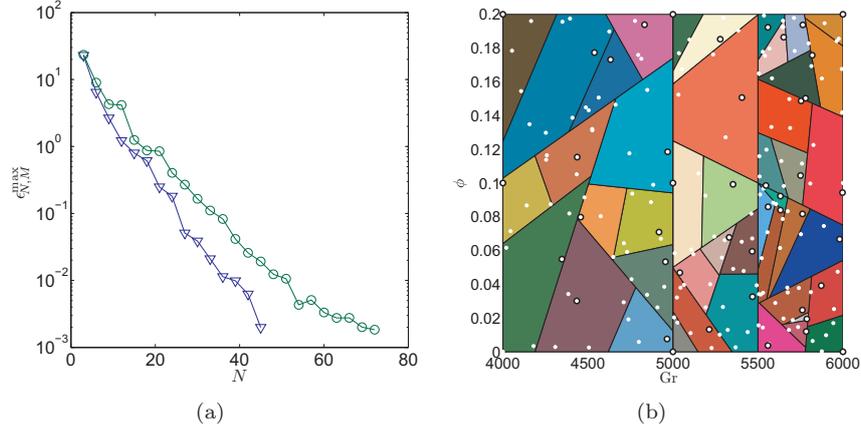


Figure 5: (a) Convergence: hp -RB (triangles) and p -type RB (circles). (b) Parameter domain partition: we show the anchor point (a circled white dot) and the Greedily selected parameters (white dots) in each subdomain; note that, within a subdomain, parameters are often selected more than once by the POD/Greedy algorithm.

truth FE space, we choose $X^{\mathcal{N}} = Z^{\mathcal{N}} \times W^{\mathcal{N}}$ of dimension $\mathcal{N} = 7248$, where $Z^{\mathcal{N}}$ denotes a discretely divergence-free \mathbb{P}_2 space for the velocity (developed from the $\mathbb{P}_2 - \mathbb{P}_1$ Taylor-Hood velocity-pressure approximation) and $W^{\mathcal{N}}$ is a standard \mathbb{P}_2 FE space for the temperature. For further details on the formulation of this problem see [12].

We note that for the computational results for this problem, we consider a “relative $L^2(\Omega)$ error bound” version of Algorithm 2 and hence Algorithm 3. To obtain a benchmark for comparison we first perform a standard (p -type) POD/Greedy computation: we specify $\epsilon = 2 \cdot 10^{-3}$ for the target tolerance, $\Delta N = 3$ for the number of POD modes to include at each Greedy iteration, $\mu^* = (6000, 0)$ for the initial parameter value, and a train sample Ξ_{train} of size 200. In this case Algorithm 2 terminates for $N_{\max} = \tilde{N}_{\max} = 72$. Recall that in the quadratically nonlinear case the POD/Greedy terminates when the *nominal* error bound reaches the prescribed tolerance.

We then perform an hp -POD/Greedy computation: we specify $\epsilon_{\text{tol}}^1 = 1.2$, $\epsilon_{\text{tol}}^2 = 2 \cdot 10^{-3}$, $\bar{N} = 45$, $\Delta N = 3$, $\hat{\mu}_{(1)}^0 = (6000, 0)$, and a train sample $\Xi_{\text{train},(1)}$ of size 9. In this case Algorithm 2 terminates after generation of $M = 45$ subdomains with $N_{\max} = 45$. The maximum relative RB $L^2(\Omega)$ error bound

An hp -RB Method for Parabolic Equations

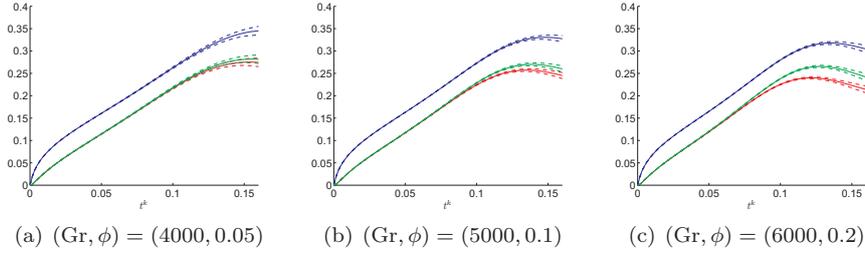


Figure 6: The RB outputs $s_{N,1}(t^k; \mu)$ (red, solid line), $s_{N,2}(t^k; \mu)$ (blue, solid line), $s_{N,3}(t^k; \mu)$ (green, solid line), and associated error bars (dashed lines) as functions of time for three values of μ .

$\epsilon_{N,M}^{\max}$ (over the train samples) over all subdomains for the hp -RB approximation as well as for the p -type RB approximation are shown in Figure 5(a). As in the linear case, the hp approach trades reduced N for increased M . We show the hp -RB parameter domain partition in Figure 5(b).

In Figure 6 we show for $N = 45$ the RB output approximations to the three outputs (5.2) for three parameter values $(Gr, \phi) = (4000, 0.05)$, $(Gr, \phi) = (5000, 0.1)$, and $(Gr, \phi) = (6000, 0.2)$. We also indicate the corresponding error bars $[s_{N,j}^k(\mu) - \Delta_{N,s_j}^k(\mu), s_{N,j}^k(\mu) + \Delta_{N,s_j}^k(\mu)]$, $1 \leq k \leq K$, $1 \leq j \leq 3$, in which the true result $s_j^{N^k}$ must reside. We recall that the RB output error bounds Δ_{N,s_j} are obtained as the product of the RB field error bound Δ_N and the dual norm of the output functional (Eq. (3.7)).¹⁰ We remark that the accuracy of these hp -RB outputs is comparable with the accuracy of the p -type RB outputs since the hp -POD/Greedy and p -type POD/Greedy calculations terminate for the same specified tolerance. Note that time is measured in diffusive units and hence the final time of 0.16 is sufficient to observe (at these Gr) significant nonlinear effects.

The standard (p -type) RB method yields a significant Online speedup relative to the expensive Boussinesq truth FE solves (one truth solve requires 239 seconds); nevertheless, these p -type RB computations are still rather expensive due to the $O(N^4)$ complexity of the RB error bound for quadratically nonlinear problems. The hp -POD/Greedy method of this paper provides a significant additional speedup in the hp -RB Online stage due to the direct control of N_{\max}

¹⁰We note that $\sup_{v \in X^N} \frac{\ell_n(v; \mu)}{\|v\|_{L^2}} = \frac{1}{\mu_1 |D_n|} \sup_{v \in X^N} \frac{\int_{D_n} v}{\|v\|_{L^2}} \leq \frac{1}{\mu_1 \sqrt{|D_n|}} \sqrt{\frac{\int_{D_n} v^2}{\int_{\Omega} v^2}} \leq \frac{1}{\mu_1 \sqrt{|D_n|}}$.

and hence reduction in N : Online p -type RB calculation of the output and error bound with $N = 72$ basis functions requires 6.48 seconds, whereas Online hp -RB calculation of the output and error bound with $M = 45$ subdomains and $N = 45$ requires only 0.845 seconds. Of course Offline cost and Online storage are larger for the hp -RB than for the standard RB: the Offline stage requires about 69 hours for the hp -RB and only about 5.2 hours for the standard RB; the Online Dataset requires 2.3 Gbytes for the hp -RB and only 481 Mbytes for the standard RB.

We finally note that the additional splitting step (“insurance”) was invoked for ten subdomains for the hp -POD/Greedy computation, and the average of N_{\max, B^m} , $1 \leq m \leq M$, is 40.1. This suggests that ϵ_{tol}^1 in this case was reasonably well chosen with respect to the target \bar{N} .

A Proof of Lemma 1

From (4.5), (4.7), and (4.9) we obtain (4.13) with

$$c_a = Q_a \max_{2 \leq q \leq Q_a} (\gamma^q L_a^q). \quad (\text{A.1})$$

From (4.4), (4.8), and (4.10) we obtain (4.14) with

$$c_f = Q_f \max_{1 \leq q \leq Q_f} (\|f^q\|_{L^2} L_f^q). \quad (\text{A.2})$$

B Proof of Lemma 2

From (2.4) with $v = u^{\mathcal{N}^k}(\mu)$ we obtain

$$\begin{aligned} & \frac{1}{\Delta t} m(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu)) + a(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu); \mu) \\ &= \frac{1}{\Delta t} m(u^{\mathcal{N}^{k-1}}(\mu), u^{\mathcal{N}^k}(\mu)) + f(u^{\mathcal{N}^k}(\mu); \mu). \end{aligned} \quad (\text{B.1})$$

We next recall Young’s inequality $AB \leq (A^2/\kappa + \kappa B^2)/2$ (for $A, B, \kappa \in \mathbb{R}$). For the first term on the right we first invoke the Cauchy-Schwarz inequality and then Young’s inequality for $A = m(u^{\mathcal{N}^{k-1}}(\mu), u^{\mathcal{N}^{k-1}}(\mu))^{1/2}$, $B = m(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu))^{1/2}$, and $\kappa = 1$ to obtain

$$\begin{aligned} m(u^{\mathcal{N}^{k-1}}(\mu), u^{\mathcal{N}^k}(\mu)) &\leq m(u^{\mathcal{N}^{k-1}}(\mu), u^{\mathcal{N}^{k-1}}(\mu))^{1/2} m(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu))^{1/2} \\ &\leq \frac{1}{2} (m(u^{\mathcal{N}^{k-1}}(\mu), u^{\mathcal{N}^{k-1}}(\mu)) + m(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu))). \end{aligned} \quad (\text{B.2})$$

For the second term on the right we first invoke boundedness of $f(\cdot; \mu)$ and then Young's inequality with $A = \|f(\cdot; \mu)\|_{X'}$, $B = \|u^{\mathcal{N}^k}(\mu)\|_X$, and $\kappa = \alpha(\mu)$ to obtain

$$\begin{aligned} f(u^{\mathcal{N}^k}(\mu); \mu) &\leq \|f(\cdot; \mu)\|_{X'} \|u^{\mathcal{N}^k}(\mu)\|_X \leq \frac{1}{2} \left(\frac{\|f(\cdot; \mu)\|_{X'}^2}{\alpha(\mu)} + \alpha(\mu) \|u^{\mathcal{N}^k}(\mu)\|_X^2 \right) \\ &\leq \frac{1}{2} \left(\frac{\|f(\cdot; \mu)\|_{X'}^2}{\alpha(\mu)} + a(u^{\mathcal{N}^k}(\mu), u^{\mathcal{N}^k}(\mu); \mu) \right), \quad (\text{B.3}) \end{aligned}$$

where the last step follows from coercivity of $a(\cdot, \cdot; \mu)$. We combine (B.2) and (B.3) with (B.1), invoke (4.11), substitute k' for k , and sum over k' to obtain (4.16).

C Proof of Lemma 3

From linearity of (3.1) we obtain, for $1 \leq k \leq K$,

$$\begin{aligned} &\frac{1}{\Delta t} m(\Delta u_N^k - \Delta u_N^{k-1}, v) + a(\Delta u_N^k, v; \mu_2) \\ &= f(v; \mu_1) - f(v; \mu_2) + a(u_N^k(\mu_1), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1), \quad \forall v \in X_N. \end{aligned} \quad (\text{C.1})$$

Next, from Lemma 1 we obtain

$$\begin{aligned} &\frac{1}{\Delta t} m(\Delta u_N^k - \Delta u_N^{k-1}, v) + a(\Delta u_N^k, v; \mu_2) \\ &= f(v; \mu_1) - f(v; \mu_2) + a(u_N^k(\mu_1), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1) \\ &\leq c_f |\mu_1 - \mu_2| \|v\|_X + c_a |\mu_1 - \mu_2| \|u_N^k(\mu_1)\|_X \|v\|_X. \end{aligned} \quad (\text{C.2})$$

For the first term on the right we invoke Young's inequality for $A = c_f |\mu_1 - \mu_2|$, $B = \|v\|_X$, and $\kappa = \underline{\alpha}/2$ to note that

$$\begin{aligned} c_f |\mu_1 - \mu_2| \|v\|_X &\leq \frac{1}{2} \left(\frac{2c_f^2}{\underline{\alpha}} |\mu_1 - \mu_2|^2 + \frac{\underline{\alpha}}{2} \|v\|_X^2 \right) \\ &\leq \frac{c_f^2}{\underline{\alpha}} |\mu_1 - \mu_2|^2 + \frac{1}{4} a(v, v; \mu_2), \quad (\text{C.3}) \end{aligned}$$

where the second inequality follows from coercivity of $a(\cdot, \cdot; \mu_2)$. For the second term on the right we invoke Young's inequality for $A = c_a |\mu_1 - \mu_2| \|u_N^k(\mu_1)\|_X$, $B = \|v\|_X$, and $\kappa = \underline{\alpha}/2$ to note that

$$\begin{aligned} c_a |\mu_1 - \mu_2| \|u_N^k(\mu_1)\|_X \|v\|_X &\leq \frac{1}{2} \left(\frac{2c_a^2}{\underline{\alpha}} |\mu_1 - \mu_2|^2 \|u_N^k(\mu_1)\|_X^2 + \frac{\underline{\alpha}}{2} \|v\|_X^2 \right) \\ &\leq \frac{c_a^2}{\underline{\alpha}^2} |\mu_1 - \mu_2|^2 a(u_N^k(\mu_1), u_N^k(\mu_1); \mu_1) + \frac{1}{4} a(v, v; \mu_2), \end{aligned} \quad (\text{C.4})$$

where the second inequality follows from coercivity of $a(\cdot, \cdot; \mu)$. With (C.2), (C.3), and (C.4) we obtain for $v = \Delta u_N^k$,

$$\begin{aligned} m(\Delta u_N^k, \Delta u_N^k) + \frac{\Delta t}{2} a(\Delta u_N^k, \Delta u_N^k; \mu_2) &\leq m(\Delta u_N^{k-1}, \Delta u_N^k) \\ &\quad + \frac{\Delta t}{\underline{\alpha}^2} |\mu_1 - \mu_2|^2 \left(\underline{\alpha} c_f^2 + c_a^2 a(u_N^k(\mu_1), u_N^k(\mu_1); \mu_1) \right). \end{aligned} \quad (\text{C.5})$$

For the first term on the right we note by the Cauchy-Schwarz inequality and Young's inequality for $A = m(\Delta u_N^{k-1}, \Delta u_N^{k-1})^{1/2}$, $B = m(\Delta u_N^k, \Delta u_N^k)^{1/2}$, and $\kappa = 1$ that

$$\begin{aligned} m(\Delta u_N^{k-1}, \Delta u_N^k) &\leq m(\Delta u_N^{k-1}, \Delta u_N^{k-1})^{1/2} m(\Delta u_N^k, \Delta u_N^k)^{1/2} \\ &\leq \frac{1}{2} m(\Delta u_N^{k-1}, \Delta u_N^{k-1}) + \frac{1}{2} m(\Delta u_N^k, \Delta u_N^k). \end{aligned} \quad (\text{C.6})$$

Hence

$$\begin{aligned} m(\Delta u_N^k, \Delta u_N^k) - m(\Delta u_N^{k-1}, \Delta u_N^{k-1}) + \Delta t a(\Delta u_N^k, \Delta u_N^k; \mu_2) \\ \leq \frac{2\Delta t}{\underline{\alpha}^2} |\mu_1 - \mu_2|^2 \left(\underline{\alpha} c_f^2 + c_a^2 a(u_N^k(\mu_1), u_N^k(\mu_1); \mu_1) \right). \end{aligned} \quad (\text{C.7})$$

We now substitute k' for k and sum over k' to obtain

$$\|\Delta u_N^k\|_{\mu_2}^2 \leq \frac{2}{\underline{\alpha}^2} |\mu_1 - \mu_2|^2 \left(\underline{\alpha} c_f^2 t^k + c_a^2 \Delta t \sum_{k'=1}^k a(u_N^{k'}(\mu_1), u_N^{k'}(\mu_1); \mu_1) \right). \quad (\text{C.8})$$

We finally note that $\Delta t \sum_{k'=1}^k a(u_N^{k'}(\mu_1), u_N^{k'}(\mu_1); \mu_1) \leq \|\Delta u_N^k(\mu_1)\|_{\mu_1}^2$. Hence, by Lemma 2 we obtain (4.17) for \tilde{C} given in (4.18).

D Proof of Lemma 4

From linearity of (3.1) we obtain, for $1 \leq k \leq K$,

$$\begin{aligned} & \frac{1}{\Delta t} m(\Delta u_N^k - \Delta u_N^{k-1}, v) + a(\Delta u_N^k, v; \mu_2) \\ &= f(v; \mu_1) - f(v; \mu_2) + a(u_N^k(\mu_1), v; \mu_2) - a(u_N^k(\mu_1), v; \mu_1), \quad \forall v \in X_N. \end{aligned} \quad (\text{D.1})$$

We choose $v = (\Delta u_N^k - \Delta u_N^{k-1})/\Delta t \in X_N$ and obtain

$$\begin{aligned} & \frac{1}{\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + \frac{1}{\Delta t} a(\Delta u_N^k, \Delta u_N^k - \Delta u_N^{k-1}; \mu_2) \\ &= \frac{1}{\Delta t} f(\Delta u_N^k - \Delta u_N^{k-1}; \mu_1) - \frac{1}{\Delta t} f(\Delta u_N^k - \Delta u_N^{k-1}; \mu_2) \\ &+ \frac{1}{\Delta t} a(u_N^k(\mu_1), \Delta u_N^k - \Delta u_N^{k-1}; \mu_2) - \frac{1}{\Delta t} a(u_N^k(\mu_1), \Delta u_N^k - \Delta u_N^{k-1}; \mu_1), \end{aligned} \quad (\text{D.2})$$

for all $v \in X_N$. From Lemma 1 we obtain

$$\begin{aligned} & \frac{1}{\Delta t} f(\Delta u_N^k - \Delta u_N^{k-1}; \mu_1) - \frac{1}{\Delta t} f(\Delta u_N^k - \Delta u_N^{k-1}; \mu_2) \\ & \leq \frac{c_f}{\Delta t} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2| \end{aligned} \quad (\text{D.3})$$

and

$$\begin{aligned} & \frac{1}{\Delta t} a(u_N^k(\mu_1), \Delta u_N^k - \Delta u_N^{k-1}; \mu_2) - \frac{1}{\Delta t} a(u_N^k(\mu_1), \Delta u_N^k - \Delta u_N^{k-1}; \mu_1) \\ & \leq \frac{c_a}{\Delta t} \|u_N^k(\mu_1)\|_X \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2|. \end{aligned} \quad (\text{D.4})$$

We thus obtain

$$\begin{aligned} & \frac{1}{\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + \frac{1}{\Delta t} a(\Delta u_N^k, \Delta u_N^k - \Delta u_N^{k-1}; \mu_2) \\ & \leq \frac{c_f}{\Delta t} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2| \\ & \quad + \frac{c_a}{\Delta t} \|u_N^k(\mu_1)\|_X \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2|. \end{aligned} \quad (\text{D.5})$$

We now recall from (4.5) that $a(\cdot, \cdot; \mu) = a^1(\cdot, \cdot) + a_{\text{II}}(\cdot, \cdot; \mu)$. We may thus write

$$\begin{aligned} & \frac{1}{\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + \frac{1}{\Delta t} a^1(\Delta u_N^k, \Delta u_N^k) \\ & \leq \frac{1}{\Delta t} a^1(\Delta u_N^k, \Delta u_N^{k-1}) + \frac{1}{\Delta t} |a_{\text{II}}(\Delta u_N^k, \Delta u_N^k - \Delta u_N^{k-1}; \mu_2)| \\ & \quad + \frac{c_f}{\Delta t} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2| \\ & \quad + \frac{c_a}{\Delta t} \|u_N^k(\mu_1)\|_X \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2} |\mu_1 - \mu_2|. \end{aligned} \quad (\text{D.6})$$

Next, we apply the Cauchy-Schwarz inequality to the first term on the right and continuity to the second term on the right; we then apply Young's inequality to each term on the right to obtain

$$\begin{aligned} & \frac{1}{\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + \frac{1}{\Delta t} a^1(\Delta u_N^k, \Delta u_N^k) \\ & \leq \frac{1}{2\Delta t} \left(a^1(\Delta u_N^k, \Delta u_N^k) + a^1(\Delta u_N^{k-1}, \Delta u_N^{k-1}) \right) \\ & \quad + \frac{\bar{\gamma}}{2} \left(\frac{1}{3\bar{\gamma}\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + 3\bar{\gamma} \|\Delta u_N^k\|_X^2 \right) \\ & \quad + \frac{1}{2} \left(\frac{1}{3\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + 3c_f^2 |\mu_1 - \mu_2|^2 \right) \\ & \quad + \frac{1}{2} \left(\frac{1}{3\Delta t^2} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + 3c_a^2 \|u_N^k(\mu_1)\|_X^2 |\mu_1 - \mu_2|^2 \right), \end{aligned} \quad (\text{D.7})$$

or

$$\begin{aligned} & \frac{1}{\Delta t} \|\Delta u_N^k - \Delta u_N^{k-1}\|_{L^2}^2 + a^1(\Delta u_N^k, \Delta u_N^k) - a^1(\Delta u_N^{k-1}, \Delta u_N^{k-1}) \\ & \leq 3\bar{\gamma}^2 \Delta t \|\Delta u_N^k\|_X^2 + 3|\mu_1 - \mu_2|^2 \left(c_f^2 \Delta t + c_a^2 \Delta t \|u_N^k(\mu_1)\|_X^2 \right). \end{aligned} \quad (\text{D.8})$$

We then substitute k' for k and sum over k' to obtain

$$\begin{aligned} & \frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 + a^1(\Delta u_N^k, \Delta u_N^k) \\ & \leq 3\bar{\gamma}^2 \sum_{k'=1}^k \Delta t \|\Delta u_N^{k'}\|_X^2 + 3|\mu_1 - \mu_2|^2 \left(c_f^2 t^k + c_a^2 \Delta t \sum_{k'=1}^k \|u_N^{k'}(\mu_1)\|_X^2 \right). \end{aligned} \quad (\text{D.9})$$

Finally, we first invoke coercivity of $a(\cdot, \cdot; \mu)$, and then Lemmas 2 and 3 to obtain

$$\begin{aligned}
 & \frac{1}{\Delta t} \sum_{k'=1}^k \|\Delta u_N^{k'} - \Delta u_N^{k'-1}\|_{L^2}^2 + a^1(\Delta u_N^k, \Delta u_N^k) \\
 & \leq \frac{3\bar{\gamma}^2}{\underline{\alpha}} \Delta t \sum_{k'=1}^k a(\Delta u_N^{k'}, \Delta u_N^{k'}; \mu_2) \\
 & \quad + \frac{3}{\underline{\alpha}} |\mu_1 - \mu_2|^2 \left(\underline{\alpha} c_f^2 t^k + c_a^2 \Delta t \sum_{k'=1}^k a(u_N^{k'}(\mu_1), u_N^{k'}(\mu_1); \mu_1) \right) \\
 & \leq |\mu_1 - \mu_2|^2 \frac{3}{\underline{\alpha}^2} \left(\bar{\gamma}^2 \underline{\alpha} \tilde{C}^2 + t^k \underline{\alpha}^2 c_f^2 + t^k c_a^2 \max_{\mu \in \mathcal{D}} \|f(\cdot; \mu)\|_{X'} \right). \quad (\text{D.10})
 \end{aligned}$$

The desired result thus follows since $a^1(\Delta u_N^k, \Delta u_N^k) \geq 0$.

Acknowledgments

This work was supported by the Norwegian University of Science and Technology, AFOSR Grant No. FA9550-07-1-0425, and OSD/AFOSR Grant No. FA9550-09-1-0613.

Bibliography

- [1] B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16:525–528, 1978.
- [2] D. Amsellem, J. Cortial, and C. Farhat. On-Demand CFD-Based Aeroelastic Predictions Using a Database of Reduced-Order Bases and Models. In *47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition, Orlando, FL*, January 2009. AIAA Paper 2009-800.
- [3] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

- [4] J. L. Eftang, M. A. Grepl, and A. T. Patera. A posteriori error bounds for the empirical interpolation method. *Comptes Rendus Mathematique*, 348(9-10):575 – 579, 2010.
- [5] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An “*hp*” certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200, 2010.
- [6] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An *hp* certified reduced basis method for parametrized parabolic partial differential equations. In J. S. Hesthaven and E. M. Rønquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, volume 76 of *Lecture Notes in Computational Science and Engineering*, pages 179–187. Springer Berlin Heidelberg, 2011.
- [7] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005.
- [8] B. Haasdonk, M. Dihlmann, and M. Ohlberger. A Training Set and Multiple Bases Generation Approach for Parametrized Model Reduction Based on Adaptive Grids in Parameter Space. Technical Report 28, SRC SimTech, 2010.
- [9] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
- [10] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique*, 345(8):473 – 478, 2007.
- [11] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey. libMesh: A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations. *Engineering with Computers*, 22(3–4):237–254, 2006.
- [12] D. J. Knezevic, N. C. Nguyen, and AT Patera. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, *accepted*, 2010, August.

- [13] D. J. Knezevic and A. T. Patera. A certified reduced basis method for the fokker–planck equation of dilute polymeric fluids: Fene dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793–817, 2010.
- [14] D. J. Knezevic and J. W. Peterson. A high-performance parallel implementation of the certified reduced basis method. *Computer Methods in Applied Mechanics and Engineering (submitted)*, 2010.
- [15] N. C. Nguyen, G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes: Application to real-time bayesian parameter estimation. In Biegler, Birosa, Ghattas, Heinkenschloss, Keyes, Mallick, Marzouk, Tenorio, Waanders, and Willcox, editors, *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 151–177. John Wiley & Sons, Ltd, 2010.
- [16] N. C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous burgers’ equation. *Calcolo*, 46:157–185, 2009.
- [17] A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18:455–462, 1980.
- [18] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations*, volume 23 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1994.
- [19] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [20] L. Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.

PAPER 4

**A TWO-STEP CERTIFIED REDUCED BASIS
METHOD**

J. L. EFTANG, D. B. P. HUYNH, D. J. KNEZEVIC, AND A. T. PATERA

*Accepted in
Springer Journal of Scientific Computing,
2011*

A TWO-STEP CERTIFIED REDUCED BASIS METHOD

J. L. EFTANG*, D. B. P. HUYNH[†], D. J. KNEZEVIC[†], AND A. T. PATERA[†]

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

*[†]Department of Mechanical Engineering,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA*

Abstract

In this paper we introduce a two-step Certified Reduced Basis (RB) method. In the first step we construct from an expensive finite element “truth” discretization of dimension \mathcal{N} an intermediate RB model of dimension $N \ll \mathcal{N}$. In the second step we construct from this intermediate RB model a *derived* RB (DRB) model of dimension $M \leq N$. The construction of the DRB model is effected at cost $\mathcal{O}(N)$ and in particular at cost independent of \mathcal{N} ; subsequent evaluation of the DRB model may then be effected at cost $\mathcal{O}(M)$. The DRB model comprises both the DRB output *and* a rigorous *a posteriori* error bound for the error in the DRB output with respect to the *truth* discretization.

The new approach is of particular interest in two contexts: *focus calculations* and *hp-RB approximations*. In the former the new approach serves to reduce online cost, $M \ll N$: the DRB model is restricted to a slice or subregion of a larger parameter domain associated with the intermediate RB model. In the latter the new approach enlarges the class of problems amenable to *hp*-RB treatment by a significant reduction in offline (precomputation) cost: in the development of the *hp* parameter domain partition and associated “local” (now derived) RB models the finite element truth is replaced by the intermediate RB model. We present numerical results to illustrate the new approach.

1 Introduction

The Certified Reduced Basis (RB) method is a computational and mathematical framework for model order reduction of parameter dependent partial differential

equations (PDEs). In particular, the RB method provides rapid and certifiable computation of linear functional outputs — such as average field values or average fluxes — associated with the solution to the PDE for any set of input parameter values that configure the PDE in terms of (say) applied forces, material properties, geometry, or boundary conditions. The RB method is of interest in two particular contexts: *real-time* — such as parameter estimation [23] and optimal control [13] — and *many-query* — such as multiscale [3, 20] or stochastic simulation [4]. In these contexts, a computational preprocessing (offline) stage is typically justified. Early contributions to the RB methodology include [1, 24, 25]. For a review of these as well as more recent contributions, see [26].

Given any input parameter value from a predefined parameter domain, the RB field approximation is a Galerkin-optimal linear combination of N precomputed highly accurate (“truth”) \mathcal{N} -degree-of-freedom Finite Element (FE) snapshots of the solution to the PDE associated with N judiciously chosen parameter values. The RB output approximation is then evaluated as a linear functional of the RB field approximation. When the solution depends smoothly on the parameters an accurate RB approximation may be computed based on rather few precomputed snapshots: $N \ll \mathcal{N}$. Moreover, a rigorous *a posteriori* RB output error bound for the difference between the truth output and RB output may also be developed.

The efficiency of the RB method in the real-time and many-query contexts is effected through an *offline-online* computational strategy. The RB *offline* stage comprises FE snapshot selection and computation. This stage may be expensive — \mathcal{N} -dependent — but is performed only once as preprocessing. The RB *online* stage comprises evaluation of the RB output *and* RB output error bound for any given input parameter value. This stage is inexpensive — \mathcal{N} -independent — and may thus be effected in real-time and many-query contexts. The keys to the \mathcal{N} -independent online stage are efficient *construction–evaluation* computational procedures that link the offline and online stages through a stored dataset of size independent of \mathcal{N} . These procedures also provide efficient and exhaustive exploration of the parameter domain in the offline selection of optimal FE snapshots through a Greedy sampling algorithm.

In this paper we introduce a two-step Certified RB method. In the first step we construct from an expensive FE truth discretization of dimension \mathcal{N} an intermediate RB model of dimension $N \ll \mathcal{N}$. In the second step we construct from this intermediate RB model a *derived* RB (DRB) model of dimension $M \leq N$. The construction of the DRB model is effected at cost $\mathcal{O}(N)$ and in particular at cost independent of \mathcal{N} ; subsequent evaluation of the DRB model may then be effected at cost $\mathcal{O}(M)$. The DRB model comprises both the DRB

output *and* a rigorous *a posteriori* error bound for the error in the DRB output with respect to the *truth* discretization.

The DRB model is defined over a parameter *subdomain* (typically a sub-region or submanifold of the original parameter domain associated with the underlying intermediate RB model) and hence typically M can be chosen significantly smaller than N ; the DRB model thus enables an additional speedup. The key innovations of this paper are efficient DRB precomputation — the *construction cost* of the DRB model is \mathcal{N} -independent — and rigorous and efficient *a posteriori* bounds for the error in the DRB approximation — the error may be bounded rigorously with respect to the \mathcal{N} -complexity FE truth at *evaluation cost* independent of \mathcal{N} and N .

The notion of two-step model order reduction has been considered in earlier works, albeit in different contexts and with different emphasis than our approach here. In [29], a “Fourier model reduction method” for large (non-parametric) control problems is presented. The Fourier method is first applied to the original equation in order to construct an “intermediate order” reduced system; a computationally more intensive reduction method, such as balanced truncation [22], may then be applied to this intermediate order system. A two-step strategy is also pursued in [18], where a Krylov subspace method is followed by balanced truncation in the context of circuit component design.

In this paper, we consider parametric model order reduction in two contexts in which our new approach is of particular interest:

Focus calculations. We consider the case in which we require many (or real-time) RB output evaluations in a parameter subdomain or submanifold $\mathcal{D}' \subset \mathcal{D}$. For an accurate approximation over this smaller parameter subdomain, a smaller DRB model may be sufficient and hence provide faster output computation compared to the standard RB alternative. Applications include parameter estimation and in particular Bayesian inference [23] and frequentistic validation [14], as well as visualization or indeed design or optimization of an RB output or RB error bound over a 1-parameter or 2-parameter slice of the full parameter domain.

***hp*-RB approximation.** The *hp*-RB method was recently introduced in [7]. This approach provides an online speedup of the RB approximation through an optimal and automatic partition (*h*-refinement) of the full parameter domain \mathcal{D} into K parameter subdomains $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$. A standard RB model of dimension N^k is then constructed for each parameter subdomain (*p*-refinement);

Paper 4

presumably we may choose $N^k \ll N$ since each “local” approximation space is invoked for a smaller range of parameter values. However, although the online speedup associated with an hp -RB approximation may be significant, the offline cost can be rather large: the dimension reduction effected within each subdomain does not balance the number of parameter subdomains in terms of total offline computational cost. Thus, in particular, the hp -RB offline stage requires

$$N_{\text{total}} = \sum_{k=1}^K N^k > N \quad (1.1)$$

truth FE snapshot computations in total.

With the new two-step approach introduced in this paper, we replace the N_{total} expensive offline FE truth snapshot computations in the hp -RB offline stage with much less expensive RB snapshot approximations; we then replace the standard RB model associated with each parameter subdomain by a DRB model. Through this hp -DRB approach, we may significantly reduce the hp -RB offline cost and hence broaden the class of problems amenable to hp -RB treatment. We include a summary of the hp -RB method in Section 5.1.

We may also pursue a mixed approach (for focus calculations or hp -RB approximations), in which the underlying intermediate RB model is in fact an hp -RB model. However, in particular with an hp -DRB approach, there is in this case a delicate balance in the offline stage between additional FE snapshot computations (for the underlying hp -RB model) and faster hp -RB snapshot computation (for the DRB models). We do not consider this mixed approach further in this paper.

The paper is organized as follows. We introduce in Section 2 the problem statement as well as notation required later; we also introduce two model problems to which we shall apply the new method. We introduce in Section 3 the new two-step approximation scheme; we discuss the (Greedy) construction of the RB and DRB approximation spaces, *a posteriori* error estimation, and the associated (construction-evaluation) computational procedures. We consider in Section 4 and Section 5 the new approach in the context of focus calculations and in the context of hp -RB approximations, respectively. In each context we discuss the associated offline-online computational decoupling, and we present numerical results for our two model problems; for all our numerical results we use `rb00mit` [21], which is an RB plugin for the open source FE library `libMesh` [19]. Finally, in Section 6, we summarize the paper and discuss some areas of future work.

2 Problem Statement

2.1 Abstract Framework

We consider linear elliptic second order partial differential equations. For simplicity in the exposition of our approach we consider the formulation only for real-valued fields, however the extension to complex fields is straightforward and in fact in our second model problem (Helmholtz acoustic horn) we present results for this complex case. We introduce the spatial domain $\Omega \subset \mathbb{R}^d$ ($d = 1, 2, 3$); we shall denote a particular spatial point $x \in \Omega$ as $x = (x_{(1)}, \dots, x_{(d)})$. We further specify the function spaces $L_2(\Omega) = \{v : \int_{\Omega} v^2 < \infty\}$, $H_1(\Omega) = \{|\nabla v| \in L^2(\Omega)\}$, and $H_0^1(\Omega) = \{v \in H^1(\Omega), v|_{\partial\Omega} = 0\}$; we then introduce the space X^e associated with the exact solutions of the parametrized PDE as $H_0^1(\Omega) \subseteq X^e \subseteq H^1(\Omega)$. We next introduce a parameter domain $\mathcal{D} \subset \mathbb{R}^P$; we shall denote a particular parameter value $\mu \in \mathcal{D}$ as $\mu = (\mu_{(1)}, \dots, \mu_{(P)})$.

We next introduce a parametrized bilinear form a and a parametrized linear functional f such that for any parameter value $\mu \in \mathcal{D}$, $a(\cdot, \cdot; \mu) : X^e \times X^e \rightarrow \mathbb{R}$ is coercive and continuous over X^e , and $f(\cdot; \mu) : X^e \rightarrow \mathbb{R}$ is bounded over X^e . We also introduce an X^e -bounded linear output functional $\ell : X^e \rightarrow \mathbb{R}$ which we for simplicity assume is parameter independent. We shall further assume that a and f admit parametrically affine expansions

$$a(\cdot, \cdot; \mu) = \sum_{q=1}^{Q_a} a^q(\cdot, \cdot) \Theta_a^q(\mu), \quad (2.1)$$

$$f(\cdot; \mu) = \sum_{q=1}^{Q_f} f^q(\cdot) \Theta_f^q(\mu), \quad (2.2)$$

respectively, where $Q_a \leq Q$, $Q_f < Q$, and Q is finite and relatively small. The assumptions (2.1) and (2.2) accommodate the construction-evaluation computational procedures which we shall discuss in detail in Section 3.4. However, we note that these assumptions may be relaxed by the Empirical Interpolation Method [2, 5, 9], which in the non-affine case serves to construct affine expansions that are good approximations to the non-affine forms.

We denote by $\bar{\mu} \in \mathcal{D}$ a fixed ‘‘reference’’ parameter value; we then introduce the X -inner product and the associated X -norm for any $v, w \in X^e$ as

$$(w, v)_X = \frac{1}{2}(a(w, v; \bar{\mu}) + a(v, w; \bar{\mu})), \quad \|v\|_X = \sqrt{(v, v)_X}, \quad (2.3)$$

respectively (more generally we may consider any inner product with induced norm equivalent to $\|\cdot\|_X$). We further introduce the coercivity and continuity constants of a ,

$$\alpha^e(\mu) = \inf_{v \in X^e} \frac{a(v, v; \mu)}{\|v\|_X^2}, \quad \gamma^e(\mu) = \sup_{v \in X^e} \sup_{w \in X^e} \frac{a(v, w; \mu)}{\|v\|_X \|w\|_X}, \quad (2.4)$$

respectively.

We may now introduce the abstract formulation of the exact problem. Given any parameter value $\mu \in \mathcal{D}$, find $u^e(\mu) \in X^e$ such that

$$a(u^e(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X^e, \quad (2.5)$$

and then evaluate the exact output of interest as

$$s^e(\mu) = \ell(u^e(\mu)). \quad (2.6)$$

We next introduce a high-fidelity truth FE approximation space $X \equiv X^{\mathcal{N}} \subset X^e$ of finite dimension \mathcal{N} . We may then introduce the truth FE discretization of (2.5)–(2.6): given any $\mu \in \mathcal{D}$, find $u(\mu) \in X$ such that

$$a(u(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X, \quad (2.7)$$

and then evaluate the truth output of interest as

$$s(\mu) = \ell(u(\mu)). \quad (2.8)$$

We shall assume that X is chosen rich enough (and thus \mathcal{N} large enough) that, for any $\mu \in \mathcal{D}$, the error between the exact solution $u^e(\mu)$ and the truth approximation $u(\mu)$ is negligible at the desired level of numerical accuracy for the RB approximation; the RB approximation shall be built upon, and the RB error shall be bounded with respect to, this FE truth approximation.

We now introduce the coercivity and continuity constants of a with respect to X ,

$$\alpha(\mu) = \inf_{v \in X} \frac{a(v, v; \mu)}{\|v\|_X^2}, \quad \gamma(\mu) = \sup_{v \in X} \sup_{w \in X} \frac{a(v, w; \mu)}{\|v\|_X \|w\|_X}, \quad (2.9)$$

respectively; for our *a posteriori* error estimators, we shall also require a coercivity lower bound α_{LB} : $0 < \alpha_{\text{LB}}(\mu) \leq \alpha(\mu)$, for all $\mu \in \mathcal{D}$. An efficient computational procedure for the computation of a coercivity lower bound is possible through the Successive Constraint Method (SCM) [16, 17, 26].

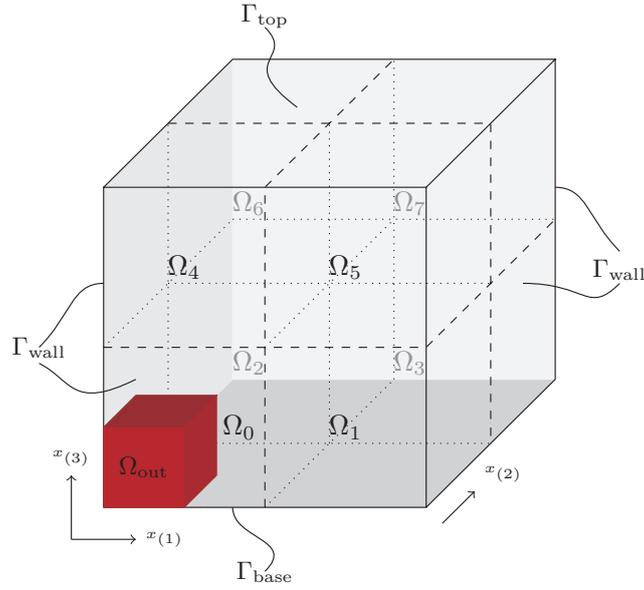


Figure 1: The thermal block.

The RB method [26] provides an acceleration of the truth (2.7)–(2.8) by the construction of an approximation space of low dimension $N \ll \mathcal{N}$. This space is optimized for the particular problem at hand, and thus provides accurate approximations despite the relatively low cost. The DRB method, which is the focus of this paper, further accelerates the RB approximation in contexts such as focus calculations and *hp*-RB approximations by the construction of an approximation space *derived* from an intermediate RB approximation space. This DRB approximation space is tailored to a parameter subdomain or submanifold of the original parameter domain, and is of even lower dimension $M \leq N$.

2.2 Model Problems

2.2.1 A 3D Thermal Block

We introduce here a “thermal block” linear elliptic model problem. We specify the spatial domain (the thermal block) $\Omega = (0, 1)^3$, which is partitioned into

eight subblocks

$$\Omega_0 = (0, 0.5) \times (0, 0.5) \times (0, 0.5), \quad (2.10)$$

$$\Omega_1 = (0.5, 1) \times (0, 0.5) \times (0, 0.5), \quad (2.11)$$

$$\Omega_2 = (0, 0.5) \times (0.5, 1) \times (0, 0.5), \quad (2.12)$$

$$\Omega_3 = (0.5, 1) \times (0.5, 1) \times (0, 0.5), \quad (2.13)$$

$$\Omega_4 = (0, 0.5) \times (0, 0.5) \times (0.5, 1), \quad (2.14)$$

$$\Omega_5 = (0.5, 1) \times (0, 0.5) \times (0.5, 1), \quad (2.15)$$

$$\Omega_6 = (0, 0.5) \times (0.5, 1) \times (0.5, 1), \quad (2.16)$$

$$\Omega_7 = (0.5, 1) \times (0.5, 1) \times (0.5, 1), \quad (2.17)$$

as shown in Figure 1. We shall consider the nondimensionalized temperature $u^e(\mu)$ in Ω . We specify unity (inward) heat flux on the floor $\Gamma_{\text{base}} = \{x \in \partial\Omega : x_{(3)} = 0\}$; we specify thermal insulation $\partial u^e / \partial n = 0$ on the walls $\Gamma_{\text{wall}} = \{x \in \partial\Omega : x_{(1)} = 0 \text{ or } x_{(1)} = 1\} \cup \{x \in \partial\Omega : x_{(2)} = 0 \text{ or } x_{(2)} = 1\}$ (here n denotes the outward normal unit vector); and we specify zero temperature $u^e = 0$ on the top $\Gamma_{\text{top}} = \{x \in \partial\Omega : x_{(3)} = 1\}$. We require continuity of the temperature and of the heat flux across interior boundaries. We next specify the parameter domain $\mathcal{D} = [0.5, 2]^7$; the thermal conductivity in the seven subblocks Ω_i , $1 \leq i \leq 7$, is given by $\mu_{(i)}$, $1 \leq i \leq 7$. The thermal conductivity in Ω_0 is equal to unity.

We now specify the exact space $X^e = \{v \in H^1(\Omega) : v|_{\Gamma_{\text{top}}} = 0\}$. We then specify, for all $\mu \in \mathcal{D}$ and for any $w, v \in X^e$, the bilinear form and linear functional

$$a(w, v; \mu) = \int_{\Omega_0} \nabla w \cdot \nabla v + \sum_{i=1}^7 \mu_{(i)} \int_{\Omega_i} \nabla w \cdot \nabla v, \quad (2.18)$$

$$f(v; \mu) = \int_{\Gamma_{\text{base}}} v, \quad (2.19)$$

respectively. We also specify, for any $v \in X^e$, the output functional

$$\ell(v) = \frac{1}{|\Omega_{\text{out}}|} \int_{\Omega_{\text{out}}} v, \quad (2.20)$$

where $\Omega_{\text{out}} = (0, 0.25) \times (0, 0.25) \times (0, 0.25)$ and $|\Omega_{\text{out}}| = 0.25^3$ is the size of Ω_{out} . The exact weak formulation for the temperature $u^e(\mu)$ in Ω is then given by (2.5); the exact output $s^e(\mu) = \ell(u^e(\mu))$ corresponds to the average temperature over Ω_{out} . We note that our affine assumptions (2.1)–(2.2) hold

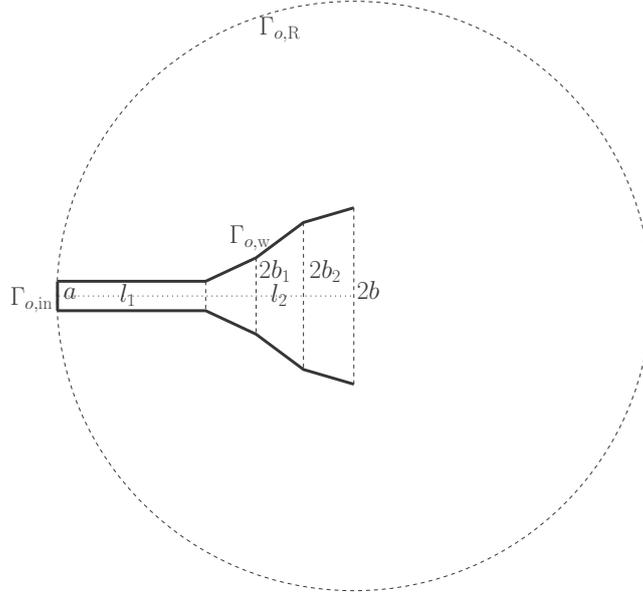


Figure 2: The acoustic horn.

for $Q_a = 8$ and $Q_f = 1$. We choose for this problem the reference parameter $\bar{\mu} = (1, 1, 1, 1, 1, 1, 1) \in \mathbb{R}^7$; thus $(w, v)_X = \int_{\Omega} \nabla w \cdot \nabla v$.

For our numerical results of Section 4.2.1 (focus calculations) and Section 5.4.1 (hp -RB approximations) we use for our truth calculations a standard $\mathbb{P}_1(\Omega)$ FE approximation space $X = X^{\mathcal{N}}$ of dimension $\mathcal{N} = 9261$, which is deemed sufficiently rich. The truth FE formulation of the problem is then given by (2.7). We note that with our choice of inner product our problem is coercive with a coercivity lower bound given for all $\mu \in \mathcal{D}$ by $\alpha_{\text{LB}}(\mu) = \min\{1, \mu_{(1)}, \dots, \mu_{(7)}\}$. (In fact here $\alpha_{\text{LB}}(\mu) \leq \alpha^e(\mu) (\leq \alpha(\mu))$.)

2.2.2 A 2D Acoustic Horn

We introduce here a Helmholtz linear elliptic model problem, first proposed in [27]. We specify a parametrized two-dimensional domain $\Omega_o(\mu) \subset \mathbb{R}^2$, which corresponds to a parameter dependent acoustic horn inside a truncated circular domain as shown in Figure 2. (The subscript $_o$ denotes an “original” quantity;

for our computational procedures we consider $\Omega_o(\mu)$ as the image of a parameter independent “reference” domain under a piecewise affine mapping.) The horn consists of a straight channel of width $a = 1$ and length $l_1 = 3$, followed by a flared section of length $l_2 = 5$. The outlet is of width $2b = 10$. The expansion channel is divided into 3 sections of equal length $5/3$. The wall $\Gamma_{o,W}$ of the symmetric expansion channel is modeled as a piecewise linear function; the heights of the sections, b_1 and b_2 , are considered as our (geometric) parameters. The domain is truncated at the circle $\Gamma_{o,R}$ of radius $R = 12.5$ centered slightly away from the outlet of the horn.

We shall consider the nondimensionalized (complex) pressure $u_o^e(\mu)$ in $\Omega_o(\mu)$; in this subsection $i = \sqrt{-1}$. We specify a source, $\frac{\partial u_o^e(\mu)}{\partial n_o} + i\mu_{(3)}u_o^e(\mu) = 2i\mu_{(3)}$, at the inlet $\Gamma_{o,in}$; we specify a first order (Sommerfeld) radiation boundary condition, $\frac{\partial u_o^e(\mu)}{\partial n_o} + \left(i\mu_{(3)} + \frac{1}{2R}\right)u_o^e(\mu) = 0$, at the radiation boundary $\Gamma_{o,R}$; and we specify a Neumann boundary condition, $\partial u_o^e(\mu)/\partial n_o = 0$, on the horn wall $\Gamma_{o,W}$. We next specify the parameter domain $\mathcal{D} = [1.0, 1.8] \times [1.8, 2.5] \times [0, 2]$; we denote a particular parameter value as $\mu = (\mu_{(1)}, \mu_{(2)}, \mu_{(3)}) = (b_1, b_2, k) \in \mathcal{D}$, where k is the nondimensional frequency or wave number.

We now define our complex space $X_o^e = \{v = v_R + iv_I : v_R \in H^1(\Omega_o(\mu)), v_I \in H^1(\Omega_o(\mu))\}$. Let \bar{v} denote the complex conjugate of v . We then specify, for all $\mu \in \mathcal{D}$ and for any $w, v \in X_o^e$, the sesquilinear form and anti-linear functional

$$\begin{aligned} a_o(w, v; \mu) &= (1 + i\epsilon) \int_{\Omega_o(\mu)} \nabla w \cdot \nabla \bar{v} - \mu_{(3)} \int_{\Omega_o(\mu)} w \bar{v} \\ &\quad + \int_{\Gamma_{o,in}} w \bar{v} + \left(\frac{1}{2R} + i\mu_{(3)}\right) \int_{\Gamma_{o,R}} w \bar{v}, \end{aligned} \quad (2.21)$$

$$f_o(v; \mu) = 2i\mu_{(3)} \int_{\Gamma_{o,in}} \bar{v}, \quad (2.22)$$

respectively. Here $\epsilon = 0.001$ represent a small dissipation in the medium. We also specify, for any $v \in X_o^e$, the output functional

$$\ell_o(v) = \int_{\Gamma_{o,in}} \bar{v}; \quad (2.23)$$

the output thus corresponds to a measurement of the pressure at the inlet $\Gamma_{o,in}$.

We then apply a domain decomposition technique (see [26]) to represent the bilinear and linear forms in our usual affine expansions: we divide $\Omega_o(\mu)$ into

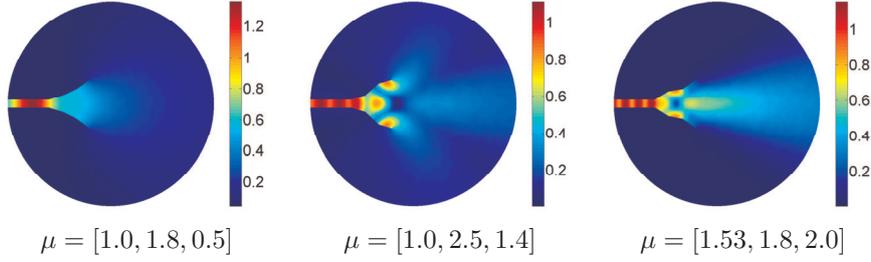


Figure 3: The magnitude of the pressure field in $\Omega_o(\mu)$ for different parameter values.

20 subdomains and consider each subdomain as the image of a parameter independent “reference subdomain” under an affine transformation; we denote the union of these reference subdomains by Ω ($\equiv \Omega_o(\bar{\mu})$, where $\bar{\mu} = (1.4, 2.15, 0)$). We also introduce a space X^e such that any $v \in X^e$ maps to $v_o \in X_o^e$ through our piecewise affine transformation. The exact weak formulation for the pressure $u^e(\mu) \in X^e$ in the reference domain Ω is then given by a complex version of (2.5). Furthermore, through the domain decomposition technique we obtain complex versions of (2.1) and (2.2) for $Q_a = 25$ and $Q_f = 1$, respectively. We finally define, for all $w, v \in X^e$, our X -inner product for this problem as

$$(w, v)_X = \int_{\Omega} \nabla w \cdot \nabla \bar{v} + \int_{\Omega} w \bar{v}. \quad (2.24)$$

For our numerical results in Section 4.2.2 (focus calculations) and Section 5.4.2 (hp -RB approximations) we use for our truth calculations a standard $\mathbb{P}_1(\Omega)$ FE approximation space $X = X^{\mathcal{N}} \subset X^e$ of dimension $\mathcal{N} = 30108$, which is sufficiently accurate for our choice of frequency range. For purposes of illustration we show in Figure 3 three solution fields corresponding to different parameter values.

Although with the dissipation (and radiation) condition this problem is in fact coercive, it is preferable to consider for our *a posteriori* error estimators not a coercivity constant lower bound but rather an inf-sup constant lower bound β_{LB} : $0 < \beta_{\text{LB}}(\mu) \leq \beta(\mu)$. Here,

$$\beta(\mu) = \inf_{w \in X} \sup_{v \in X} \frac{|a(w, v; \mu)|}{\|w\|_X \|v\|_X}, \quad (2.25)$$

for all $\mu \in \mathcal{D}$, where $|\cdot|$ denotes complex modulus. Typically, this positive inf-sup lower bound is constructed by a natural norm version of the SCM procedure [16]. However, in this paper, for simplicity¹ we choose β_{LB} to be a constant: the minimum of the SCM lower bound over a dense set in \mathcal{D} . Admittedly, this choice will compromise both sharpness (since we invoke a minimum) and rigor (since this minimum is taken over a subset of \mathcal{D}) of our *a posteriori* error bound.

3 The Certified Derived Reduced Basis Method

In this section we introduce the new two-step RB method. For simplicity our development here is for coercive linear elliptic equations with real-valued fields. However, the extension to non-coercive equations and complex fields — required for our Helmholtz acoustic horn model problem — is straightforward.

3.1 Derived RB approximation

We introduce the intermediate (standard) RB approximation space $X_N \subset X$ of dimension $N \ll \mathcal{N}$. The space X_N is spanned by solutions of (2.7) for judiciously chosen (see Section 3.3) parameter values $\mu_1 \in \mathcal{D}, \dots, \mu_N \in \mathcal{D}$,

$$X_N \equiv \text{span}\{u(\mu_1), \dots, u(\mu_N)\} \equiv \text{span}\{\zeta_1, \dots, \zeta_N\}; \quad (3.1)$$

here, $\{\zeta_1, \dots, \zeta_N\}$ denotes an X -orthonormal basis for X_N , obtained through (say) a modified Gram-Schmidt procedure.²

We may then introduce the RB approximation: given any $\mu \in \mathcal{D}$, find $u_N(\mu) \in X_N$ such that

$$a(u_N(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_N, \quad (3.2)$$

and then evaluate the RB output approximation as

$$s_N(\mu) = \ell(u_N(\mu)). \quad (3.3)$$

¹The natural-norm SCM procedure in [16] has a multi-parameter domain structure different from the multi-parameter domain structure of the *hp*-RB approach considered in this paper; a streamlined merger of these approaches is the subject of future work.

²In the modified Gram-Schmidt procedure we compute $\tilde{\zeta}_i = u(\mu_i) - \sum_{n=1}^{i-1} (\zeta_n, u(\mu_i))_X \zeta_n$, $2 \leq i \leq N$, in an iterative fashion in order to preserve numerical stability in finite precision as described in [8]. Here $\zeta_1 = u(\mu_1)/\|u(\mu_1)\|_X$ and $\zeta_i = \frac{\tilde{\zeta}_i}{\|\tilde{\zeta}_i\|_X}$, $2 \leq i \leq N$.

We now introduce a parameter subdomain or submanifold $\mathcal{D}' \subset \mathcal{D}$ to which the DRB model shall be specifically tailored. In the context of focus calculations, we wish to speed up evaluation of the RB solution, RB output, and RB error bound for any parameter value in the subdomain $\mathcal{D}' \subset \mathcal{D}$; in the context of hp -RB approximations, we wish to speedup evaluation of the RB solution, RB output, and RB error bound for any parameter value in \mathcal{D} through a partition of \mathcal{D} into *many* (K) subdomains $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$ subdomains.³ With regard to the hp -RB approximation, \mathcal{D}' denotes in this section any of the K subdomains \mathcal{V}^k , $1 \leq k \leq K$; the hp -RB approximation is discussed in greater detail in Section 5.

We introduce the DRB approximation space $X_{N,M} \subset X_N$ of dimension $M \leq N$. The space $X_{N,M}$ is spanned by solutions of (3.2) for judiciously chosen (see Section 3.3) parameter values $\mu'_1 \in \mathcal{D}', \dots, \mu'_M \in \mathcal{D}'$,

$$X_{N,M} \equiv \text{span}\{u_N(\mu'_1), \dots, u_N(\mu'_M)\} \equiv \text{span}\{\psi_1, \dots, \psi_M\}. \quad (3.4)$$

Here, $\{\psi_1, \dots, \psi_M\}$ denotes an X -orthonormal basis for $X_{N,M}$, obtained through a Gram-Schmidt procedure; however we note that in practice, we shall not require the explicit (\mathcal{N} -dependent) computation of ψ_1, \dots, ψ_M . The computational link between the intermediate and derived RB models will be discussed later in Section 3.4.

We may now finally introduce the DRB approximation: given any $\mu \in \mathcal{D}'$, find $u_{N,M}(\mu) \in X_{N,M}$ such that

$$a(u_{N,M}(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_{N,M}, \quad (3.5)$$

and then evaluate the DRB output approximation as

$$s_{N,M}(\mu) = \ell(u_{N,M}(\mu)). \quad (3.6)$$

3.2 *A Posteriori* Error Estimation

We first recall the *a posteriori* error estimator for the (standard) RB approximation [26]. We define the residual

$$r_N(\cdot; \mu) = a(u_N(\mu), \cdot; \mu) - f(\cdot; \mu) \in X'; \quad (3.7)$$

³As described in the introduction, the DRB provides an *offline* (and not online *per se*) speedup of the hp -RB approximation. This offline speedup enlarges the class of problems amenable to RB treatment. However, this offline speedup may also accomodate larger K — smaller subdomains — and thus implicitly a speedup of the hp -RB online cost.

we then introduce the Riesz representation of the residual, $\mathcal{R}_N(\mu) \in X$, which satisfies

$$(\mathcal{R}_N(\mu), v)_X = r_N(v; \mu), \quad \forall v \in X. \quad (3.8)$$

We may then define the RB error bound as⁴

$$\Delta_N(\mu) = \frac{\|\mathcal{R}_N(\mu)\|_X}{\alpha_{\text{LB}}(\mu)}. \quad (3.9)$$

We may readily demonstrate that $\|u(\mu) - u_N(\mu)\|_X \leq \Delta_N(\mu)$: We first note that the error $e_N(\mu) = u(\mu) - u_N(\mu)$ satisfies

$$a(e_N(\mu), v; \mu) = r_N(v; \mu), \quad \forall v \in X. \quad (3.10)$$

We then choose $v = e_N(\mu)$ and invoke (3.8) to obtain

$$a(e_N(\mu), e_N(\mu); \mu) = (\mathcal{R}_N(\mu), e_N(\mu))_X. \quad (3.11)$$

We apply coercivity to the left hand side and the Cauchy-Schwarz inequality to the right hand side to obtain

$$\alpha_{\text{LB}}(\mu) \|e_N(\mu)\|_X^2 \leq \|\mathcal{R}_N(\mu)\|_X \|e_N(\mu)\|_X, \quad (3.12)$$

from where we readily derive (3.9). We shall discuss the computation of $\Delta_N(\mu)$ — in particular the dual norm of the residual $\|\mathcal{R}_N(\mu)\|_X$ — in Section 3.4; however we note here that we may in the RB evaluation stage, for any given $\mu \in \mathcal{D}$, compute $\Delta_N(\mu)$ at cost $\mathcal{O}(Q^2 N^2)$ — independently of the truth complexity \mathcal{N} .

The *a posteriori* error estimator for the DRB approximation is very similar. We define the residual

$$r_{N,M}(\cdot; \mu) = a(u_{N,M}(\mu), \cdot; \mu) - f(\cdot; \mu) \in X'; \quad (3.13)$$

we then introduce the Riesz representation of the residual, $\mathcal{R}_{N,M}(\mu) \in X$, which satisfies

$$(\mathcal{R}_{N,M}(\mu), v)_X = r_{N,M}(v; \mu), \quad \forall v \in X. \quad (3.14)$$

⁴We note that for our Helmholtz acoustic problem the RB error bound is given as in (3.9) with the coercivity constant lower bound α_{LB} replaced by an inf-sup constant lower bound β_{LB} .

We then define the error bound

$$\Delta_{N,M}(\mu) = \frac{\|\mathcal{R}_{N,M}(\mu)\|_X}{\alpha_{\text{LB}}(\mu)}, \quad (3.15)$$

for which we may show that $\|u(\mu) - u_{N,M}(\mu)\|_X \leq \Delta_N(\mu)$ by arguments analogous to (3.10)–(3.12). We emphasize that $\Delta_{N,M}(\mu)$ bounds the error in the DRB approximation with respect to the truth upon which the *intermediate* RB model is built. We shall discuss the computation of $\Delta_{N,M}(\mu)$ in detail in Section 3.4; however we note here that we may in the DRB evaluation stage, for any given $\mu \in \mathcal{D}'$, compute $\Delta_{N,M}(\mu)$ at cost $\mathcal{O}(Q^2 M^2)$ — independently of the truth complexity \mathcal{N} and the RB complexity N .

For our sampling algorithm which we discuss in the next section we shall also require a bound for the error in the DRB approximation with respect to the intermediate RB approximation. We introduce the Riesz representation of the DRB residual in the RB space, $\tilde{\mathcal{R}}_{N,M}(\mu) \in X_N$, which satisfies

$$(\tilde{\mathcal{R}}_{N,M}(\mu), v)_X = r_{N,M}(v), \quad \forall v \in X_N. \quad (3.16)$$

We then define the error bound

$$\tilde{\Delta}_{N,M}(\mu) = \frac{\|\tilde{\mathcal{R}}_{N,M}(\mu)\|_X}{\alpha_{\text{LB}}(\mu)}; \quad (3.17)$$

for which we may show that $\|u_N(\mu) - u_{N,M}(\mu)\|_X \leq \tilde{\Delta}_{N,M}(\mu)$ by arguments analogous to (3.10)–(3.12).

Finally, we note that we may readily develop error bounds for the RB (or DRB) output approximation. For example, for any $\mu \in \mathcal{D}$,

$$|s(\mu) - s_{N,M}(\mu)| = |\ell(u(\mu) - u_{N,M}(\mu))| \quad (3.18)$$

$$\leq \sup_{v \in X} \frac{\ell(v)}{\|v\|_X} \|u(\mu) - u_{N,M}(\mu)\|_X \quad (3.19)$$

$$\leq \|\ell\|_{X'} \Delta_{N,M}(\mu). \quad (3.20)$$

3.3 Greedy Parameter Sampling

For the construction of both the intermediate RB space X_N and the DRB space $X_{N,M}$, we invoke a Greedy parameter sampling procedure [26, 28], which we now discuss.

Algorithm 1 $X_{N_{\max}} = \text{Greedy}^{\text{RB}}(\mu_1, \epsilon_{\text{tol}}^{\text{RB}})$

```

 $N \leftarrow 1$ 
 $X_N = \text{span}\{u(\mu_N)\}$ 
 $\epsilon_N^{\max} = \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}}} \Delta_N(\mu)$ 
while  $\epsilon_N^{\max} > \epsilon_{\text{tol}}^{\text{RB}}$  do
   $N \leftarrow N + 1$ 
   $\mu_N = \arg \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}}} \Delta_{N-1}(\mu)$ 
   $X_N = X_{N-1} \oplus \text{span}\{u(\mu_N)\}$ 
   $\epsilon_N^{\max} = \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}}} \Delta_N(\mu)$ 
end while
 $N_{\max} = N$ 

```

We first consider the construction of the intermediate RB approximation space. We introduce a training set $\Xi_{\text{train}}^{\mathcal{D}} \subset \mathcal{D}$ of finite cardinality $|\Xi_{\text{train}}^{\mathcal{D}}|$ which shall serve as a computational surrogate for \mathcal{D} . We then introduce as Algorithm 1 the Greedy^{RB} sampling procedure. For a specified tolerance $\epsilon_{\text{tol}}^{\text{RB}}$ and an initial parameter value $\mu_1 \in \mathcal{D}$, Algorithm 1 returns a space $X_{N_{\max}} \subset X$ of dimension N_{\max} such that $\Delta_{N_{\max}}(\mu) \leq \epsilon_{\text{tol}}^{\text{RB}}$ for all $\mu \in \Xi_{\text{train}}^{\mathcal{D}}$. We typically choose $\Xi_{\text{train}}^{\mathcal{D}}$ “dense” and hence we may anticipate that $\Delta_{N_{\max}}(\mu) \leq \epsilon_{\text{tol}}^{\text{RB}}$ for most $\mu \in \mathcal{D}$. We note that due to the hierarchical structure of the spaces — $X_1 \subset \dots \subset X_{N_{\max}}$ — we may readily extract spaces of dimension $N < N_{\max}$ from $X_{N_{\max}}$.

We next consider the construction of the DRB approximation space. We introduce a training set $\Xi_{\text{train}}^{\mathcal{D}'} \subset \mathcal{D}' (\subset \mathcal{D})$ of finite cardinality $|\Xi_{\text{train}}^{\mathcal{D}'}|$ which shall serve as our computational surrogate for \mathcal{D}' . We then introduce as Algorithm 2 the Greedy^{DRB} sampling procedure. For a specified tolerance $\epsilon_{\text{tol}}^{\text{DRB}}$, a desired intermediate RB space (upon which the DRB space is built) dimension $N \leq N_{\max}$, and an initial parameter value $\mu'_1 \in \mathcal{D}'$, Algorithm 2 returns a space $X_{N, M_{\max}} \subseteq X_N$ of dimension $M_{\max} \leq N$ such that $\tilde{\Delta}_{N, M_{\max}}(\mu) \leq \epsilon_{\text{tol}}^{\text{DRB}}$ for all $\mu \in \Xi_{\text{train}}^{\mathcal{D}'}$. We note that due to the hierarchical structure of the spaces — $X_{N,1} \subset \dots \subset X_{N, M_{\max}}$ — we may readily extract spaces of dimension $M < M_{\max}$ from $X_{N, M_{\max}}$. We emphasize that Algorithm 2 is identical to Algorithm 1 except for the procedures for snapshot computation and error bound evaluation.

We note that in Algorithm 2 we invoke the error bound (3.17) with respect to the intermediate RB approximation in order to ensure convergence of the algorithm: the maximum error bound $\epsilon_{N, M}^{\max} \rightarrow 0$ as $M \rightarrow N$ and hence any specified tolerance $\epsilon_{\text{tol}}^{\text{DRB}} > 0$ will eventually be satisfied. We also note that, for

Algorithm 2 $X_{N, M_{\max}} = \text{Greedy}^{\text{DRB}}(\mu'_1, N, \epsilon_{\text{tol}}^{\text{DRB}})$

$M \leftarrow 1$
 $X_{N, M} = \text{span}\{u(\mu'_M)\}$
 $\epsilon_{N, M}^{\max} = \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}'}} \tilde{\Delta}_{N, M}(\mu)$
while $\epsilon_{N, M}^{\max} > \epsilon_{\text{tol}}^{\text{DRB}}$ **do**
 $M \leftarrow M + 1$
 $\mu'_M = \arg \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}'}} \tilde{\Delta}_{N, M-1}(\mu)$
 $X_{N, M} = X_{N, M-1} \oplus \text{span}\{u_N(\mu'_M)\}$
 $\epsilon_M^{\max} = \max_{\mu \in \Xi_{\text{train}}^{\mathcal{D}'}} \tilde{\Delta}_{N, M}(\mu)$
end while
 $M_{\max} = M$

any $\mu \in \Xi_{\text{train}}^{\mathcal{D}'}$, the error in the DRB approximation with respect to the truth can be bounded as

$$\begin{aligned} \|u(\mu) - u_{N, M}(\mu)\|_X &\leq \|u(\mu) - u_N(\mu)\|_X + \|u_N(\mu) - u_{N, M}(\mu)\|_X \\ &\leq \Delta_N(\mu) + \tilde{\Delta}_{N, M}(\mu) \leq \epsilon_N^{\max} + \epsilon_{N, M}^{\max}. \end{aligned} \quad (3.21)$$

However, we can not reduce the term ϵ_N^{\max} since we increase only M (and not N) during the Greedy^{DRB} sampling procedure. As a result we typically choose in practice $\epsilon_{\text{tol}}^{\text{DRB}} > \epsilon_{\text{tol}}^{\text{RB}}$ in order to avoid Greedy^{DRB} iterations that do not provide significant error (with respect to the truth) reduction.

We emphasize that in the online stage we bound the error in the DRB approximation with respect to the truth. We note that in practice we do not invoke $\Delta_N(\mu) + \tilde{\Delta}_{N, M}(\mu)$ (in (3.21)) as an error bound, since evaluation of $\Delta_N(\mu)$ is expensive (N -dependent). We thus invoke in the online stage the less expensive (evaluation cost depends on M , and not on N) bound $\Delta_{N, M}(\mu)$ in (3.15). We discuss computational procedures and associated computational cost next.

3.4 Construction-Evaluation Computational Procedures

The key ingredients in our computational procedures are the affine expansions (2.1) and (2.2) of a and f , respectively. The construction–evaluation procedures which we introduce here enable efficient offline–online computational procedures. We discuss application of the construction–evaluation procedures to

the offline–online decoupling for each of our two particular applications, focus calculation and hp -RB approximation, in Section 4 and Section 5, respectively.

3.4.1 Output Approximation

RB output. We first expand the RB field approximation in terms of the basis functions ζ_1, \dots, ζ_N of X_N as

$$u_N(\mu) = \sum_{n=1}^N u_{N,n}(\mu) \zeta_n. \quad (3.22)$$

With (2.1) and (2.2) we may then write (3.2) as the linear system

$$\sum_{j=1}^N u_{N,j}(\mu) \left(\sum_{q=1}^{Q_a} a^q(\zeta_j, \zeta_i) \Theta_a^q(\mu) \right) = \sum_{q=1}^{Q_f} f^q(\zeta_i) \Theta_f^q(\mu), \quad 1 \leq i \leq N, \quad (3.23)$$

in the coefficients $u_{N,j}(\mu)$, $1 \leq j \leq N$. We obtain the RB output approximation (3.3) as

$$s_N(\mu) = \ell(u_N(\mu)) = \sum_{n=1}^N u_{N,n}(\mu) \ell(\zeta_n). \quad (3.24)$$

We now identify the construction and evaluation stages. In the construction stage we compute for $1 \leq q \leq Q_a$ the “stiffness matrices” $A_N^q \equiv \{a^q(\zeta_j, \zeta_i)\} \in \mathbb{R}^{N \times N}$; we compute for $1 \leq q \leq Q_f$ the “load vectors” $F_N^q \equiv \{f^q(\zeta_i)\} \in \mathbb{R}^N$; we also compute the terms $\ell(\zeta_i)$ ($1 \leq i \leq N$) required for the output. The construction stage is performed at cost $\mathcal{O}(N^\bullet)$. In the evaluation stage, given any $\mu \in \mathcal{D}$, we evaluate $\Theta_a^q(\mu)$, $1 \leq q \leq Q_a$, and $\Theta_f^q(\mu)$, $1 \leq q \leq Q_f$, at cost $\mathcal{O}(Q)$; we then perform the two summations over q in (3.23) at cost $\mathcal{O}(Q_a N^2 + Q_f N)$, and solve the $N \times N$ linear system for the RB coefficients $u_{N,n}(\mu)$, $1 \leq n \leq N$, at cost $\mathcal{O}(N^3)$ (we must anticipate that the RB system matrix is dense). We finally evaluate the RB output approximation (3.24) at cost $\mathcal{O}(N)$.

DRB output. We first expand the basis functions ψ_1, \dots, ψ_M of $X_{N,M}$ in terms of the basis functions ζ_1, \dots, ζ_N of X_N as

$$\psi_i = \sum_{n=1}^N \kappa_{i,n} \zeta_n, \quad 1 \leq i \leq M; \quad (3.25)$$

recall that

$$\underbrace{\text{span}\{\psi_1, \dots, \psi_M\}}_{X_{N,M}} = \underbrace{\text{span}\{u_N(\mu'_1), \dots, u_N(\mu'_M)\}}_{X_{N,M}} \subset \underbrace{\text{span}\{\zeta_1, \dots, \zeta_N\}}_{X_N}, \quad (3.26)$$

where ψ_1, \dots, ψ_M is an X -orthonormal basis for $X_{N,M}$. We may obtain the coefficients $\kappa_{m,n}$, $1 \leq m \leq M$, $1 \leq n \leq N$, from the Gram-Schmidt procedure for ψ_1, \dots, ψ_M as follows. For $m = 1$, we obtain

$$\psi_1 = \frac{u_N(\mu'_1)}{\|u_N(\mu'_1)\|_X} = \frac{\sum_{n=1}^N u_{N,n}(\mu'_1) \zeta_n}{\left(\sum_{m=1}^N \sum_{n=1}^N u_{N,n}(\mu'_1) u_{N,m}(\mu'_1) \underbrace{(\zeta_m, \zeta_n)_X}_{\delta_{m,n}} \right)^{1/2}} \quad (3.27)$$

$$= \frac{\sum_{n=1}^N u_{N,n}(\mu'_1) \zeta_n}{\left(\sum_{n=1}^N (u_{N,n}(\mu'_1))^2 \right)^{1/2}} \equiv \sum_{n=1}^N \kappa_{1,n} \zeta_n, \quad (3.28)$$

where $\delta_{i,j}$ is the Kroenecker delta symbol. For $2 \leq m \leq M$, we further obtain $\psi_m = \tilde{\psi}_m / \|\tilde{\psi}_m\|_X$ where, from (3.22) and (3.25),

$$\tilde{\psi}_m = u_N(\mu'_m) - \sum_{s=1}^{m-1} (\psi_s, u_N(\mu'_m))_X \psi_s, \quad (3.29)$$

$$= \sum_{n=1}^N u_{N,n}(\mu'_m) \zeta_n - \sum_{s=1}^{m-1} \sum_{n=1}^N \sum_{k=1}^N \sum_{l=1}^N u_{N,l}(\mu'_m) \kappa_{s,k} \underbrace{(\zeta_k, \zeta_l)_X}_{\delta_{k,l}} \kappa_{s,n} \zeta_n \quad (3.30)$$

$$= \sum_{n=1}^N \left(u_{N,n}(\mu'_m) - \sum_{s=1}^{m-1} \sum_{k=1}^N u_{N,k}(\mu'_m) \kappa_{s,k} \kappa_{s,n} \right) \zeta_n \equiv \sum_{n=1}^N \tilde{\kappa}_{m,n} \zeta_n. \quad (3.31)$$

We thus identify $\kappa_{m,n} = \tilde{\kappa}_{m,n} / \|\tilde{\psi}_m\|_X$, $1 \leq n \leq N$, with

$$\|\tilde{\psi}_m\|_X = \left(\sum_{n=1}^N \sum_{k=1}^N \tilde{\kappa}_{m,n} \tilde{\kappa}_{m,k} \underbrace{(\zeta_n, \zeta_k)_X}_{\delta_{n,k}} \right)^{1/2} \quad (3.32)$$

$$= \left(\sum_{n=1}^N \tilde{\kappa}_{m,n}^2 \right)^{1/2}. \quad (3.33)$$

In practice, we do not explicitly perform this (\mathcal{N} -dependent) Gram-Schmidt procedure since we do not explicitly require the DRB basis functions ψ_m , $1 \leq$

$m \leq M$. From (3.28), (3.31), and (3.33), we readily obtain the coefficients $\kappa_{m,n}$ at cost independent of \mathcal{N} .⁵

We next expand the DRB field approximation in terms of the basis functions of $X_{N,M}$ as

$$u_{N,M}(\mu) = \sum_{m=1}^M u_{N,M,m}(\mu) \psi_m. \quad (3.34)$$

With (2.1) and (2.2) we may then write (3.5) as the linear system

$$\sum_{j=1}^M u_{N,M,j}(\mu) \left(\sum_{q=1}^{Q_a} a^q(\psi_j, \psi_i) \Theta_a^q(\mu) \right) = \sum_{q=1}^{Q_f} f^q(\psi_i) \Theta_f^q(\mu), \quad 1 \leq i \leq M \quad (3.35)$$

in the coefficients $u_{N,M,j}(\mu)$, $1 \leq j \leq M$. We obtain the DRB output approximation (3.6) as

$$s_{N,M}(\mu) = \ell(u_{N,M}(\mu)) = \sum_{m=1}^M u_{N,M,m}(\mu) \ell(\psi_m). \quad (3.36)$$

With (3.25), we note that we may write $a^q(\psi_j, \psi_i)$, $f^q(\psi_i)$, and $\ell(\psi_i)$ as

$$a^q(\psi_j, \psi_i) = \sum_{n=1}^N \kappa_{j,n} \left(\sum_{m=1}^N \kappa_{i,m} a^q(\zeta_n, \zeta_m) \right), \quad 1 \leq q \leq Q_a, \quad 1 \leq i, j \leq M, \quad (3.37)$$

$$f^q(\psi_i) = \sum_{n=1}^N \kappa_{i,n} f^q(\zeta_n), \quad 1 \leq q \leq Q_f, \quad 1 \leq i \leq M, \quad (3.38)$$

$$\ell(\psi_i) = \sum_{n=1}^N \kappa_{i,n} \ell(\zeta_n), \quad 1 \leq i \leq M, \quad (3.39)$$

respectively. We may then identify the construction and evaluation stages. In the construction stage we first obtain, for $1 \leq q \leq Q_a$, the matrices $A_{N,M}^q \equiv$

⁵In (3.28), (3.31), and (3.33) we invoke the fact that $(\zeta_k, \zeta_l)_X = \delta_{k,l}$; however this is only true in infinite precision. An improvement to the numerical stability of our approach is thus to compute and store $(\zeta_k, \zeta_l)_X$, $1 \leq k, l \leq N$; we may then obtain $\kappa_{1,n}$ from (3.27) rather than from (3.28), and $\tilde{\kappa}_{m,n}$ and $\|\tilde{\psi}_m\|_X$, $2 \leq m \leq M$, from (3.30) and (3.32) rather than from (3.31) and (3.33), respectively. Note that this approach is still \mathcal{N} -independent.

A Two-Step Certified Reduced Basis Method

$\{a^q(\psi_j, \psi_i)\} \in \mathbb{R}^{M \times M}$ from the matrices $A_N^q \in \mathbb{R}^{N \times N}$ by (3.37) at cost $\mathcal{O}(N^2M)$ through a sum factorization technique as follows: for $1 \leq q \leq Q_a$, we first compute and store (temporarily) the terms

$$\tau_{i,n}^q = \sum_{m=1}^N \kappa_{i,m} a^q(\zeta_n, \zeta_m), \quad 1 \leq i \leq M, \quad 1 \leq n \leq N, \quad (3.40)$$

at cost $\mathcal{O}(N^2M)$; we then perform the outer summation

$$a^q(\psi_j, \psi_i) = \sum_{n=1}^N \kappa_{j,n} \tau_{i,n}^q, \quad 1 \leq i, j \leq M, \quad (3.41)$$

at cost $\mathcal{O}(M^2N)$. The total cost of (3.37) is thus $\mathcal{O}(N^2M)$ (for each q) since $M \leq N$.

We next obtain, for $1 \leq q \leq Q_f$, the vectors $F_{N,M}^q \equiv \{f^q(\psi_i)\} \in \mathbb{R}^M$ from the vectors $F_N^q \in \mathbb{R}^N$ by (3.38) at cost $\mathcal{O}(MN)$; and we obtain $\ell(\psi_i)$, ($1 \leq i \leq M$) from $\ell(\zeta_i)$, ($1 \leq i \leq N$), by (3.39) at cost $\mathcal{O}(MN)$. The cost of the construction stage is thus \mathcal{N} -independent. In the evaluation stage, given any $\mu \in \mathcal{D}'$, we evaluate $\Theta_a^q(\mu)$, $1 \leq q \leq Q_a$, and $\Theta_f^q(\mu)$, $1 \leq q \leq Q_f$, at cost $\mathcal{O}(Q)$; we then perform the two summations over q in (3.35) at cost $\mathcal{O}(Q_a M^2 + Q_f M)$, and solve the $M \times M$ linear system for the DRB coefficients $u_{N,M,m}(\mu)$, $1 \leq m \leq M$, at cost $\mathcal{O}(M^3)$. We finally evaluate the DRB output approximation (3.36) at cost $\mathcal{O}(M)$.

3.4.2 A Posteriori Error Bound

We discuss here the computational procedures associated with the residual dual norms required for our *a posteriori* error estimators. We refer to [16, 17, 26] for the computational procedures associated with the coercivity (or stability factor) lower bound (the SCM).

RB residual dual norm. We now discuss the construction-evaluation procedure for the dual X -norm of the RB residual. With (2.1), (2.2), and (3.22),

we may expand (3.8) as

$$(\mathcal{R}_N(\mu), v)_X = \sum_{q=1}^{Q_f} f^q(v) \Theta_q^f(\mu) - \sum_{n=1}^N u_{N,n}(\mu) \sum_{q=1}^{Q_a} a^q(\zeta_n, v) \Theta_a^q(\mu) \quad (3.42)$$

$$= \sum_{i=1}^{\bar{N}} \phi^i(\mu) \mathcal{L}^i(v), \quad (3.43)$$

for all $v \in X$. Here $\bar{N} = Q_f + NQ_a$, and the $\mathcal{L}^i \in X'$ and $\phi^i : \mathcal{D} \rightarrow \mathbb{R}$ are defined explicitly as

$$\mathcal{L}^i = f^i, \quad 1 \leq i \leq Q_f, \quad (3.44)$$

$$\mathcal{L}^{Q_f+i+(n-1)Q_a} = a^i(\zeta_n, \cdot), \quad 1 \leq i \leq Q_a, \quad 1 \leq n \leq N, \quad (3.45)$$

$$\phi^i = \Theta_f^i, \quad 1 \leq i \leq Q_f, \quad (3.46)$$

$$\phi^{Q_f+i+(n-1)Q_a} = u_{N,n} \Theta_a^i, \quad 1 \leq i \leq Q_a, \quad 1 \leq n \leq N. \quad (3.47)$$

We then define $l^i \in X$, $1 \leq i \leq \bar{N}$, such that

$$(l^i, v)_X = \mathcal{L}^i(v), \quad \forall v \in X. \quad (3.48)$$

Hence, by linearity,

$$\mathcal{R}_N(\mu) = \sum_{i=1}^{\bar{N}} \phi^i(\mu) l^i. \quad (3.49)$$

We may now identify the construction and evaluation stages. In the construction stage we solve (3.48), $1 \leq i \leq \bar{N}$, and compute the inner products $(l^i, l^j)_X$, $1 \leq i, j \leq \bar{N}$, at cost $\mathcal{O}(N^\bullet)$. In the evaluation stage, given the RB solution coefficients for any $\mu \in \mathcal{D}$, we evaluate $\phi^i(\mu)$, $1 \leq i \leq \bar{N}$, at cost $\mathcal{O}(Q_f + Q_a N)$, and perform the summation

$$\|\mathcal{R}_N(\mu)\|_X^2 = \sum_{i=1}^{\bar{N}} \sum_{j=1}^{\bar{N}} \phi^i(\mu) \phi^j(\mu) (l^i, l^j)_X, \quad (3.50)$$

at cost $\mathcal{O}(\bar{N}^2) = \mathcal{O}(Q^2 N^2)$.

DRB residual dual norm. We next discuss the construction-evaluation procedure for the dual X -norm of the DRB residual. With (2.1), (2.2), and (3.34), we may expand (3.14) as

$$(\mathcal{R}_{N,M}(\mu), v)_X = \sum_{q=1}^{Q_f} f^q(v) \Theta_q^f(\mu) - \sum_{m=1}^M u_{N,M,m}(\mu) \sum_{q=1}^{Q_a} a^q(\psi_m, v) \Theta_a^q(\mu) \quad (3.51)$$

$$= \sum_{i=1}^{\bar{M}} \varphi^i(\mu) \mathcal{H}^i(v), \quad (3.52)$$

for all $v \in X$. Here $\bar{M} = Q_f + MQ_a$, and the $\mathcal{H}^i \in X'$ and $\varphi^i : \mathcal{D}' \rightarrow \mathbb{R}$ are defined explicitly as

$$\mathcal{H}^i = f^i, \quad 1 \leq i \leq Q_f, \quad (3.53)$$

$$\mathcal{H}^{Q_f+i+(m-1)Q_a} = a^i(\psi_m, \cdot), \quad 1 \leq i \leq Q_a, \quad 1 \leq m \leq M, \quad (3.54)$$

$$\varphi^i = \Theta_f^i, \quad 1 \leq i \leq Q_f, \quad (3.55)$$

$$\varphi^{Q_f+i+(m-1)Q_a} = u_{N,M,m} \Theta_a^i, \quad 1 \leq i \leq Q_a, \quad 1 \leq m \leq M. \quad (3.56)$$

We then define $h^i \in X$, $1 \leq i \leq \bar{M}$, such that

$$(h^i, v)_X = \mathcal{H}^i(v), \quad \forall v \in X. \quad (3.57)$$

Hence, by linearity,

$$\mathcal{R}_{N,M}(\mu) = \sum_{i=1}^{\bar{M}} \varphi^i(\mu) h^i. \quad (3.58)$$

We now note, by (3.25), that we may further expand the $\mathcal{H}^{Q_f+i+(m-1)Q_a}$ in (3.54) in terms of the intermediate RB basis $\{\zeta_n\}_{n=1}^N$ as

$$\mathcal{H}^{Q_f+i+(m-1)Q_a} = a^i(\psi_m, \cdot) = \sum_{n=1}^N \kappa_{m,n} a^i(\zeta_n, \cdot), \quad (3.59)$$

for $1 \leq i \leq Q_a$ and $1 \leq m \leq M$; thus, by linearity,

$$h^{Q_f+i+(m-1)Q_a} = \sum_{n=1}^N \kappa_{m,n} l^{Q_f+i+(n-1)Q_a}, \quad (3.60)$$

for $1 \leq i \leq Q_a$ and $1 \leq m \leq M$. We recall the definition of l^i , $1 \leq i \leq Q_f + NQ_a$, from (3.48), (3.44), and (3.45).

We next consider the inner products $(h^i, h^j)_X$, $1 \leq i, j \leq \bar{M}$. First, it is clear that

$$(h^i, h^j)_X = (l^i, l^j)_X, \quad 1 \leq i, j \leq Q_f; \quad (3.61)$$

further, we note that

$$(h^{Q_f+i+(m-1)Q_a}, h^j)_X = \sum_{n=1}^N \kappa_{m,n} (l^{Q_f+i+(n-1)Q_a}, l^j)_X, \quad (3.62)$$

$$(h^j, h^{Q_f+i+(m-1)Q_a})_X = \sum_{n=1}^N \kappa_{m,n} (l^j, l^{Q_f+i+(n-1)Q_a})_X, \quad (3.63)$$

for $1 \leq i \leq Q_a$, $1 \leq m \leq M$, and $1 \leq j \leq Q_f$; we finally note that

$$\begin{aligned} & (h^{Q_f+i+(m-1)Q_a}, h^{Q_f+j+(m'-1)Q_a})_X \\ &= \sum_{n=1}^N \kappa_{m,n} \left(\sum_{n'=1}^N \kappa_{m',n'} (l^{Q_f+i+(n-1)Q_a}, l^{Q_f+j+(n'-1)Q_a})_X \right), \end{aligned} \quad (3.64)$$

for $1 \leq i, j \leq Q_a$, $1 \leq m, m' \leq M$. The key observation here is that once $(l^i, l^j)_X$, $1 \leq i, j \leq \bar{N}$, are given from the intermediate RB construction stage, the analogous data $(h^i, h^j)_X$, $1 \leq i, j \leq \bar{M}$, for the DRB model may be obtained at cost $\mathcal{O}(N^\bullet)$ — independently of the truth complexity \mathcal{N} .

We may now identify the construction and evaluation stages. In the construction stage we obtain $(h^i, h^j)_X$, $1 \leq i, j \leq \bar{M}$, from $(l^i, l^j)_X$, $1 \leq i, j \leq \bar{N}$, by (3.61)–(3.64). The cost is dominated by the summation (3.64), for which we invoke a sum factorization technique: we first compute and store the term in parentheses for $1 \leq i, j \leq Q_a$, $1 \leq n \leq N$ and $1 \leq m' \leq M$ at cost $\mathcal{O}(Q_a^2 N^2 M)$; we then perform the outer summation (over n) for $1 \leq i, j$ and $1 \leq m, m' \leq M$ at cost $\mathcal{O}(Q_a^2 M^2 N)$. The total cost is thus $\mathcal{O}(Q_a^2 N^2 M)$ since $M \leq N$. (Direct evaluation of (3.64) requires $\mathcal{O}(Q^2 N^2 M^2)$ operations.) In particular, the DRB construction stage is \mathcal{N} -independent. In the evaluation stage, given the DRB solution coefficients for any $\mu \in \mathcal{D}'$, we evaluate $\varphi^i(\mu)$, $1 \leq i \leq \bar{M}$, at cost $\mathcal{O}(\bar{M}) = \mathcal{O}(Q_f + Q_a M)$, and perform the summation

$$\|\mathcal{R}_{N,M}(\mu)\|_X^2 = \sum_{i=1}^{\bar{M}} \sum_{j=1}^{\bar{M}} \varphi^i(\mu) \varphi^j(\mu) (h^i, h^j)_X \quad (3.65)$$

at cost $\mathcal{O}(\bar{M}^2) = \mathcal{O}(Q^2 M^2)$.

DRB residual dual X_N -norm. We next discuss the construction-evaluation procedure for the dual norm of the DRB residual with respect to the intermediate RB approximation space, $\|\tilde{\mathcal{R}}_{N,M}(\mu)\|_X$. With (2.1), (2.2), and (3.34), we may expand (3.16) as

$$(\tilde{\mathcal{R}}_{N,M}(\mu), v)_X = \sum_{q=1}^{Q_f} f^q(v) \Theta_q^f(\mu) - \sum_{m=1}^M u_{N,M,m}(\mu) \sum_{q=1}^{Q_a} a^q(\psi_m, v) \Theta_a^q(\mu) \quad (3.66)$$

$$= \sum_{i=1}^{\bar{M}} \varphi^i(\mu) \mathcal{H}^i(v), \quad (3.67)$$

for all $v \in X_N$. We then define $\tilde{h}^i \in X_N$, $1 \leq i \leq \bar{M}$, such that

$$(\tilde{h}^i, v)_X = \mathcal{H}^i(v), \quad \forall v \in X_N. \quad (3.68)$$

Hence, by linearity,

$$\tilde{\mathcal{R}}_{N,M}(\mu) = \sum_{i=1}^{\bar{M}} \varphi^i(\mu) \tilde{h}^i. \quad (3.69)$$

We next consider the inner products $(\tilde{h}^i, \tilde{h}^j)_X$, $1 \leq i, j \leq \bar{M}$. We note that $\tilde{h}^i \in X_N$ may be written as

$$\tilde{h}^i = \sum_{n=1}^N \eta_n^i \zeta_n, \quad 1 \leq i \leq \bar{M}, \quad (3.70)$$

where the coefficients $\eta_1^i, \dots, \eta_N^i$ satisfy

$$\sum_{n=1}^N \eta_n^i \underbrace{(\zeta_n, \zeta_m)_X}_{\delta_{m,n}} = \eta_m^i = \mathcal{H}^i(\zeta_m), \quad 1 \leq m \leq N, \quad (3.71)$$

thanks to the X -orthonormal basis for X_N . Hence

$$(\tilde{h}^i, \tilde{h}^j)_X = \sum_{m=1}^N \sum_{n=1}^N \eta_m^i \eta_n^j \underbrace{(\zeta_m, \zeta_n)_X}_{\delta_{m,n}} = \sum_{n=1}^N \mathcal{H}^i(\zeta_n) \mathcal{H}^j(\zeta_n), \quad (3.72)$$

for $1 \leq i, j \leq \bar{M}$.

We may now identify the construction and evaluation stages. In the construction stage we compute the inner products $(\tilde{h}^i, \tilde{h}^j)_X$, $1 \leq i, j \leq \bar{M}$, from (3.72) at cost $\mathcal{O}(N\bar{M}^2)$; note that $\mathcal{H}^i(\zeta_n)$, $1 \leq i \leq \bar{M}$, $1 \leq n \leq N$, may be evaluated from (3.53) and (3.59) at cost $\mathcal{O}(N^2\bar{M})$ since the matrices A_N^q , $1 \leq q \leq Q_a$, and vectors F_N^q , $1 \leq q \leq Q_f$, are computed and stored during the construction stage for the intermediate RB output. In the evaluation stage, given the DRB solution coefficients for any $\mu \in \mathcal{D}$, we evaluate $\varphi^i(\mu)$, $1 \leq i \leq \bar{M}$, at cost $\mathcal{O}(Q_f + Q_a M)$, and perform the summation

$$\|\tilde{\mathcal{R}}_{N,M}(\mu)\|_X^2 = \sum_{i=1}^{\bar{M}} \sum_{j=1}^{\bar{M}} \varphi^i(\mu) \varphi^j(\mu) (\tilde{h}^i, \tilde{h}^j)_X, \quad (3.73)$$

at cost $\mathcal{O}(\bar{M}^2) = \mathcal{O}(Q^2 M^2)$.

We note that as an alternative to the bound $\tilde{\Delta}_{N,M}(\mu)$ we may directly compute $\|u_N(\mu) - u_{N,M}(\mu)\|_X$ at cost $\mathcal{O}(QN^2 + N^3)$. However typically M is significantly smaller than N and thus computation of $\tilde{\Delta}_{N,M}(\mu)$ is typically less expensive than computation of $\|u_N(\mu) - u_{N,M}(\mu)\|_X$ when the bound is required for many μ as in the Greedy^{DRB} algorithm.

4 Focus Calculations

In the context of focus calculations we require many (or real-time) RB output (or RB error bound) evaluations over a parameter subset or submanifold $\mathcal{D}' \subset \mathcal{D}$. Given an intermediate RB model developed for the parameter domain \mathcal{D} , a smaller DRB model is typically sufficient over $\mathcal{D}' \subset \mathcal{D}$. This smaller DRB model may yield significant speedup compared to the standard RB alternative while preserving numerical accuracy.

4.1 Offline–Online Decomposition

We now discuss the offline-online decomposition associated with the focus calculation context. The *offline* stage is the construction of the intermediate RB model over \mathcal{D} : we perform Greedy^{RB} (Algorithm 1) for a specified initial parameter value $\mu_1 \in \mathcal{D}$ and a specified error bound tolerance $\epsilon_{\text{tol}}^{\text{RB}}$ (to be satisfied over the training set $\Xi_{\text{train}}^{\mathcal{D}} \subset \mathcal{D}$). This stage is expensive — the cost is $\mathcal{O}(N^\bullet)$ — but performed only once as preprocessing.

In the *online* stage, given a parameter subdomain or submanifold $\mathcal{D}' \subset \mathcal{D}$, we first construct the DRB model: we perform Greedy^{DRB} (Algorithm 2) for

A Two-Step Certified Reduced Basis Method

a specified initial parameter value $\mu'_1 \in \mathcal{D}'$, a specified intermediate RB space (constructed offline and upon which the DRB approximation is built) dimension $N \leq N_{\max}$, and a specified error bound tolerance $\epsilon_{\text{tol}}^{\text{DRB}}$ (to be satisfied over the training set $\Xi_{\text{train}}^{\mathcal{D}'} \subset \mathcal{D}'$). The cost of this step derives from RB snapshot computation and RB error bound preprocessing and evaluation; below $\bar{M}_{\max} \equiv Q_f + Q_a M_{\max}$.

1. RB snapshot computation. We compute M_{\max} intermediate RB snapshots of complexity N_{\max} . The cost is $\mathcal{O}(M_{\max}(QN_{\max}^2 + N_{\max}^3))$ (we must anticipate that the RB system is dense).
2. DRB construction. We obtain the parameter independent matrices and vectors associated with the DRB system at cost $\mathcal{O}(QN_{\max}^2 M_{\max})$; note that we obtain these entities directly from the respective intermediate RB entities (computed offline).
3. DRB error bound preprocessing. We must compute \bar{M}_{\max}^2 inner products (3.72) for our error bound $\tilde{\Delta}_{N,M}$ (used in the Greedy^{DRB} sampling procedure) and \bar{M}_{\max}^2 inner products (3.61)–(3.64) for our error bound $\Delta_{N,M}$ (used for DRB output certification). The total cost is $\mathcal{O}(Q^2 N_{\max}^2 M_{\max})$.
4. DRB error bound evaluation. We compute the DRB approximation and evaluate the DRB error bound $\tilde{\Delta}_{N,M}$ over the training set $\Xi_{\text{train}}^{\mathcal{D}'}$ at each Greedy^{DRB} iteration. The cost is, to leading order, $\mathcal{O}(M_{\max} |\Xi_{\text{train}}^{\mathcal{D}'}| (M_{\max}^3 + M_{\max}^2 Q^2))$.
5. DRB focus calculations. For any new parameter value $\mu \in \mathcal{D}'$ and given $1 \leq M \leq M_{\max}$, we perform DRB evaluation: computation of the DRB solution, DRB output, and DRB error bound with respect to the truth approximation at cost $\mathcal{O}(M^3 + M^2 Q^2)$.

Note that the focus calculation online stage includes the construction of the DRB model over \mathcal{D}' — steps 1-4 above. The key point is that this DRB model is built inexpensively (\mathcal{N} -independently) upon the underlying intermediate RB model; the subsequent DRB evaluation stage (step 5 above, performed many times over \mathcal{D}') is then independent of \mathcal{N} and N . As a result, in the many-query context, a DRB approach may provide significant speedup compared to the standard RB alternative.

We finally note the important role of the sum factorization invoked in (3.37) and (3.64). The complexity reduction — a factor of M — is significant in practice in particular for focus calculations since the calculations (3.37) and (3.64) are performed online.

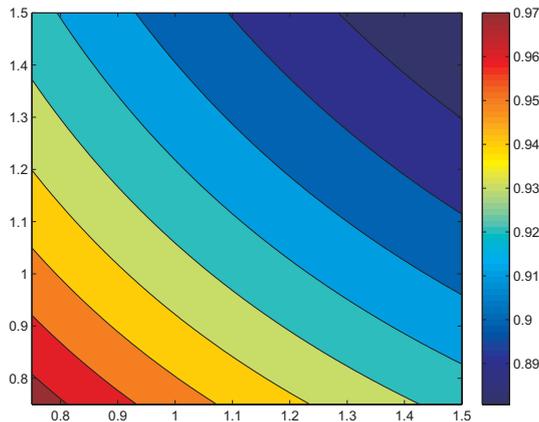


Figure 4: The RB output values on \mathcal{D}' for the thermal block; note that the DRB and standard RB outputs are indistinguishable.

4.2 Numerical Results

4.2.1 Thermal Block

We develop a DRB approximation for the thermal block problem introduced in Section 2.2.1 in order to accelerate a focus calculation. We first generate an intermediate RB approximation of dimension $N_{\max} = 96$: we perform Greedy^{RB} for $\mu_1 = (0.75, 0.75)$ and $\epsilon_{\text{tol}}^{\text{RB}} = 10^{-4}$ over a uniformly distributed random training set $\Xi_{\text{train}}^{\mathcal{D}} \subset \mathcal{D}$ of size $|\Xi_{\text{train}}^{\mathcal{D}}| = 10^4$. We then specify a two-dimensional submanifold $\mathcal{D}' \equiv [0.75, 1.5]^2 \times \{\mu_{\text{fixed}}\} \subset \mathcal{D}$, where $\mu_{\text{fixed}} = (0.7, 0.8, 0.9, 1.0, 1.1) \in \mathbb{R}^5$, and we perform RB focus calculations with this standard RB model over a 100×100 uniform grid of parameter values, $\Xi_{\text{focus}} \subset \mathcal{D}'$. The RB outputs (evaluated for each $\mu \in \Xi_{\text{train}}^{\mathcal{D}}$ via (2.20)) are shown in Figure 4; the RB output error bounds are shown in Figure 5 (top).

We then consider the corresponding DRB approach. We generate a DRB model of dimension $M_{\max} = 9$ which satisfies a tolerance $\epsilon_{\text{tol}}^{\text{DRB}} = 10^{-4}$ (with respect to the $N_{\max} = 96$ intermediate RB model) over a uniformly distributed random training set $\Xi_{\text{train}}^{\mathcal{D}'} \subset \mathcal{D}'$ of size $|\Xi_{\text{train}}^{\mathcal{D}'}| = 100$. We then calculate the DRB outputs and DRB output error bounds over Ξ_{focus} ; in this case the

A Two-Step Certified Reduced Basis Method

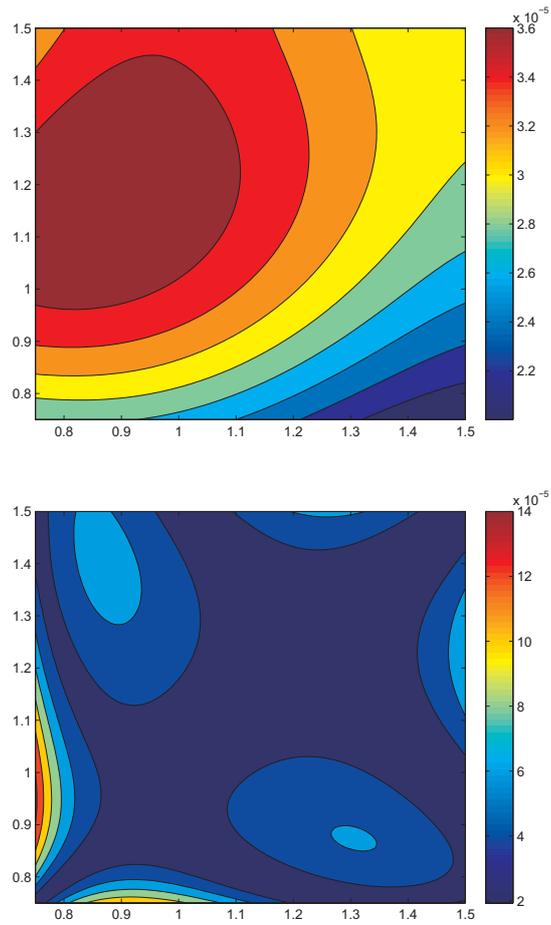


Figure 5: Standard RB output error bounds on \mathcal{D}' with respect to the truth discretization (top); and DRB output error bounds on \mathcal{D}' with respect to the truth discretization (bottom).

DRB online computation (including execution of Greedy^{DRB} and evaluation over Ξ_{focus}) is a factor of 63 faster than the standard RB alternative. Moreover, as shown in Figure 5, the maximum output error bounds (with respect to the underlying truth FE approximation) in the standard RB and derived RB approximations are $3.6 \cdot 10^{-5}$ and $14 \cdot 10^{-5}$, respectively; hence the DRB yields a significant speedup with only very mild impact on the accuracy of the approximation over \mathcal{D}' .

4.2.2 Acoustic Horn

We develop a DRB approximation for the acoustic horn problem introduced in Section 2.2.2 in order to accelerate a focus calculation. We first generate an intermediate RB approximation of dimension $N_{\text{max}} = 109$: we perform Greedy^{RB} for $\mu_1 = (1.4, 2.15, 1.0)$ and $\epsilon_{\text{tol}}^{\text{RB}} = 10^{-4}$ over a uniformly distributed random training set $\Xi_{\text{train}}^{\mathcal{D}} \subset \mathcal{D}$ of size $|\Xi_{\text{train}}^{\mathcal{D}}| = 10^4$. We then specify a one-dimensional slice $\mathcal{D}' \equiv \{\mu_{\text{fixed}}\} \times [0.5, 1.0] \subset \mathcal{D}$, where $\mu_{\text{fixed}} = (1.4, 2.2) \in \mathbb{R}^2$, and we perform RB focus calculations with this standard RB model over a uniform grid of 1000 parameter values, $\Xi_{\text{focus}} \subset \mathcal{D}'$.

We then consider the corresponding DRB approach. We generate a DRB model of dimension $M_{\text{max}} = 11$ which satisfies a tolerance $\epsilon_{\text{tol}}^{\text{DRB}} = 10^{-4}$ (with respect to the $N_{\text{max}} = 109$ intermediate RB model) over a uniformly distributed random training set $\Xi_{\text{train}}^{\mathcal{D}'} \subset \mathcal{D}'$ of size $|\Xi_{\text{train}}^{\mathcal{D}'}| = 1000$. We then calculate the DRB outputs and DRB output error bounds over Ξ_{focus} ; in this case the online computation (including execution of Greedy^{DRB} and evaluation over Ξ_{focus}) is a factor of 10 faster than the standard RB alternative. The focus calculation speedup here is less than for the thermal block because, first, \mathcal{D}' is not so “small” compared to \mathcal{D} , and second and more importantly, we perform fewer focus calculations (by a factor of 10). As shown in Figure 6, the maximum output error bounds (with respect to the underlying truth FE approximation) in the standard RB and derived RB approximations are $9.8 \cdot 10^{-5}$ and $1.2 \cdot 10^{-4}$, respectively; hence the DRB yields a significant speedup with only very mild impact on the accuracy of the approximation over \mathcal{D}' .

A Two-Step Certified Reduced Basis Method

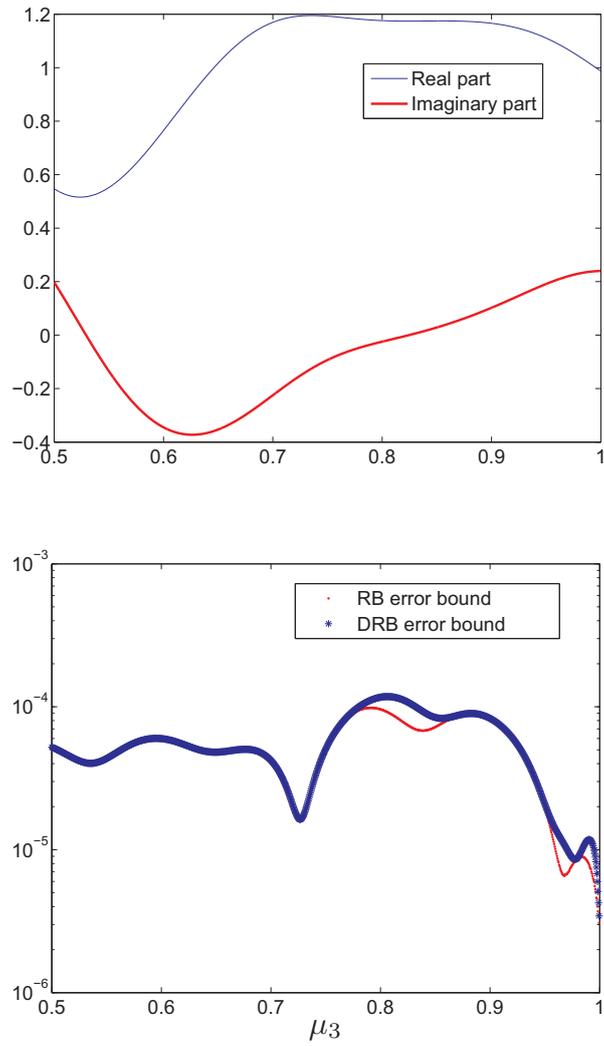


Figure 6: The RB output values on \mathcal{D}' for the acoustic horn (top), and the output error bounds on \mathcal{D}' with respect to the truth discretization for the standard RB and DRB approximations (bottom).

5 hp -RB Approximation

5.1 Summary of the hp -RB Method

The hp -RB method introduced in [7] (see also [6, 11]) provides a partition of the parameter domain \mathcal{D} into K parameter subdomains $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$; for each parameter subdomain \mathcal{V}^k , the algorithm generates an associated RB approximation space $X_{N^k}^k \subset X$ of dimension N^k spanned by truth FE snapshots associated with parameter values within \mathcal{V}^k . The approach is motivated by order reduction: we may choose the dimension N^k of the “local” space $X_{N^k}^k$ relatively small compared to the dimension N of the “global” space X_N while preserving numerical accuracy. We thus obtain significant speedup of the RB output and RB error bound evaluation. However, the offline (precomputation) cost associated with an hp -RB approach is significantly larger than the offline cost associated with the standard RB procedure, and must thus in practice be taken into consideration.

We now review the hp -RB method. We first describe the splitting procedure for an arbitrary subdomain $\mathcal{V} \subseteq \mathcal{D}$. Given $\mathcal{V} \subseteq \mathcal{D}$ and a parameter “anchor point” $\mu_1^\mathcal{V} \in \mathcal{V}$, we compute the truth FE snapshot $u(\mu_1^\mathcal{V})$ and define the one-dimensional “temporary” RB space

$$X_1^\mathcal{V} = \text{span}\{u(\mu_1^\mathcal{V})\} \quad (5.1)$$

associated with \mathcal{V} . We next introduce a finite training set $\Xi_{\text{train}}^\mathcal{V} \subset \mathcal{V}$; we then evaluate the RB error bound $\Delta_{N=1}^\mathcal{V}$ for the RB approximation associated with the space $X_{N=1}^\mathcal{V}$ (essentially (3.9) with an appropriate change of notation) for each parameter value $\mu \in \Xi_{\text{train}}^\mathcal{V}$ — essentially one iteration of the Greedy^{RB} algorithm restricted to $\mathcal{V} \subset \mathcal{D}$ — in order to identify a second parameter value

$$\mu_2^\mathcal{V} = \arg \max_{\mu \in \Xi_{\text{train}}^\mathcal{V}} \Delta_{N=1}^\mathcal{V}(\mu). \quad (5.2)$$

We then split \mathcal{V} into two subdomains $\mathcal{V}_{\text{left}} \subset \mathcal{V}$ and $\mathcal{V}_{\text{right}} \subset \mathcal{V}$ based on (Euclidean, say) distance $\|\cdot\|_2$ to the points $\mu_1^\mathcal{V}$ and $\mu_2^\mathcal{V}$: any point $\mu \in \mathcal{V}$ belongs to $\mathcal{V}_{\text{left}}$ if and only if $\|\mu - \mu_1^\mathcal{V}\|_2 \leq \|\mu - \mu_2^\mathcal{V}\|_2$; otherwise, $\mu \in \mathcal{V}$ belongs to $\mathcal{V}_{\text{right}}$. Finally, we define $\mu_1^\mathcal{V}$ as the anchor point for $\mathcal{V}_{\text{left}}$ and we define $\mu_2^\mathcal{V}$ as the anchor point for $\mathcal{V}_{\text{right}}$.

We may now describe the hp -RB method. The first step is h -refinement: We apply the splitting scheme discussed above for $\mathcal{V} = \mathcal{D}$, and then recursively for $\mathcal{V} = \mathcal{V}_{\text{left}}$ and $\mathcal{V} = \mathcal{V}_{\text{right}}$ (sketched in Figure 7 for two levels of splitting).

A Two-Step Certified Reduced Basis Method

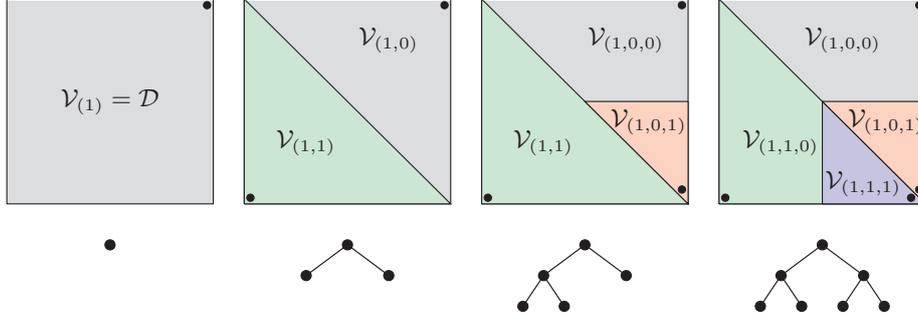


Figure 7: h -refinement partition procedure.

We terminate the splitting of a subdomain \mathcal{V} if $\max_{\mu \in \Xi_{\text{train}}^{\mathcal{V}}} \Delta_{N=1}^{\mathcal{V}}(\mu) \leq \epsilon_{\text{tol}}^h$, where ϵ_{tol}^h is a specified tolerance for the h -refinement step. The result of this hierarchical procedure is $K = K(\epsilon_{\text{tol}}^h)$ parameter subdomains $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$.

The next step is p -refinement: Greedy construction of the approximation spaces $X_{N^k}^k$, $1 \leq k \leq K$. We here choose N^k , $1 \leq k \leq K$, such that a specified tolerance $\epsilon_{\text{tol}}^p \leq \epsilon_{\text{tol}}^h$ is satisfied over training sets $\Xi_{\text{train}}^{\mathcal{V}^k} \subset \mathcal{V}^k$, $1 \leq k \leq K$. Note that this step is essentially execution of the Greedy^{RB} algorithm for $\epsilon_{\text{tol}}^{\text{RB}} = \epsilon_{\text{tol}}^p$ restricted to each subdomain $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$.

In practice, we also apply if necessary an additional splitting step (see [6]) after the p -refinement. Essentially, this step performs additional h -refinement of a subdomain if ϵ_{tol}^p is not satisfied for specified N_{max}^{hp} basis functions. The additional splitting proceeds recursively with h -refinement and p -refinement steps until ϵ_{tol}^p is satisfied for N_{max}^{hp} basis functions, thus providing for direct control of the tolerance ϵ_{tol}^p and the RB space dimension.

Thanks to the hierarchical construction of the partition, we may organize the subdomains (and associated approximation spaces) as the leaf nodes in a binary tree with Boolean flags, as illustrated in Figure 7. This tree-structure partition is important in the hp -RB online stage: the cost to determine which subdomain \mathcal{V}^{k^*} contains any given $\mu \in \mathcal{D}$ is $\mathcal{O}(\log(K))$ for K subdomains [7].

The hp -RB approximation reads as follows: first, given any $\mu \in \mathcal{D}$, determine $k^* = k^*(\mu) \in [1, K]$ and hence \mathcal{V}^{k^*} and $X_{N^{k^*}}^{k^*}$ through a binary search; given $1 \leq N \leq N_{\text{max}}^{hp}$, we write $\hat{N} = \min\{N, N^{k^*}\}$ and $X_N^{hp} = X_{\hat{N}}^{k^*}$. Then, determine

$u_N^K(\mu) \in X_N^{hp}$ such that

$$a(u_N^K(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_N^{hp}; \quad (5.3)$$

finally evaluate the hp -RB output approximation

$$s_N^K(\mu) = \ell(u_N^K(\mu)). \quad (5.4)$$

We define the hp -RB error bound as

$$\Delta_N^K(\mu) = \frac{\|\mathcal{R}_N^K(\mu)\|_X}{\alpha_{\text{LB}}(\mu)}, \quad (5.5)$$

where $\mathcal{R}_N^K(\mu) \in X$ denotes the Riesz representation of the residual

$$r_N^K(\cdot; \mu) = a(u_N^K(\mu), \cdot; \mu) - f(\cdot; \mu) \in X'. \quad (5.6)$$

We may readily show that $\|u(\mu) - u_N^K(\mu)\|_X \leq \Delta_N^K(\mu)$ by arguments analogous to (3.10)–(3.12).

5.2 DRB Modification

We now discuss the application of the two-step RB approach within the hp -RB context. We introduce a “global” intermediate RB approximation space $X_{N_{\max}}$ of dimension N_{\max} constructed by Greedy^{RB} (Algorithm 1) for a specified initial parameter value $\mu_1 \in \mathcal{D}$ and a specified error bound tolerance $\epsilon_{\text{tol}}^{\text{RB}}$ (to be satisfied over the training set $\Xi_{\text{train}}^{\mathcal{D}} \subset \mathcal{D}$). The necessary modifications to the hp -RB method discussed in the previous subsection are then as follows. First, during the h -refinement step we replace the truth snapshot $u(\mu_1^{\mathcal{V}})$ by an RB snapshot $u_{N_{\max}}(\mu_1^{\mathcal{V}}) \approx u(\mu_1^{\mathcal{V}})$; we thus replace the RB space $X_1^{\mathcal{V}}$ in (5.1) by the DRB space

$$X_{N_{\max},1}^{\mathcal{V}} = \text{span}\{u_{N_{\max}}(\mu_1^{\mathcal{V}})\}. \quad (5.7)$$

We further replace the RB error bound $\Delta_{N=1}^{\mathcal{V}}$ in (5.2) by a DRB error bound $\tilde{\Delta}_{N_{\max},M=1}^{\mathcal{V}}$ (essentially (3.17) with an appropriate change of notation). We then invoke this DRB error bound (with respect to the underlying RB approximation) to determine a second parameter value

$$\mu_2^{\mathcal{V}} = \arg \max_{\mu \in \Xi^{\mathcal{V}}} \tilde{\Delta}_{N_{\max},M=1}^{\mathcal{V}}(\mu). \quad (5.8)$$

A Two-Step Certified Reduced Basis Method

As before, $\mu_1^{\mathcal{V}}$ and $\mu_2^{\mathcal{V}}$ determine the splitting of \mathcal{V} into $\mathcal{V}_{\text{left}}$ and $\mathcal{V}_{\text{right}}$. Note that we terminate the splitting of a subdomain \mathcal{V} if $\max_{\mu \in \Xi_{\text{train}}^{\mathcal{V}}} \tilde{\Delta}_{N_{\text{max}}, M=1}^{\mathcal{V}}(\mu) \leq \epsilon_{\text{tol}}^h$; typically ϵ_{tol}^h is chosen much greater than $\epsilon_{\text{tol}}^{\text{RB}}$. As before, we apply the splitting procedure recursively until convergence; the result is $K = K(\epsilon_{\text{tol}}^h)$ subdomains $\mathcal{V}^k \subset \mathcal{D}$, $1 \leq k \leq K$.

Next, in the p -refinement step, we associate to each \mathcal{V}^k a DRB approximation space X_{N_{max}, M^k} , $1 \leq k \leq K$; the p -refinement step is thus essentially execution of Greedy^{DRB} for $\epsilon_{\text{tol}}^{\text{RB}} = \epsilon_{\text{tol}}^p$ restricted to each \mathcal{V}^k , $1 \leq k \leq K$. We typically choose ϵ_{tol}^p such that $\epsilon_{\text{tol}}^{\text{RB}} \leq \epsilon_{\text{tol}}^p < \epsilon_{\text{tol}}^h$. As before, we apply in practice an additional splitting step which provides simultaneous control over the tolerance ϵ_{tol}^p and the maximum DRB space dimension M_{max}^{hp} .

With the modifications above, the hp -DRB approximation reads as follows: first, given any $\mu \in \mathcal{D}$, determine $k^* = k^*(\mu)$ and hence \mathcal{V}^{k^*} and $X_{N_{\text{max}}, M^{k^*}}$ through a binary search; given $1 \leq M \leq M_{\text{max}}^{hp}$, we write $\hat{M} = \min\{M, M^{k^*}\}$ and $X_{N_{\text{max}}, M}^{hp} = X_{N_{\text{max}}, \hat{M}}^{k^*}$. Then, determine $u_{N_{\text{max}}, M}^K(\mu) \in X_{N_{\text{max}}, M}^{hp}$ such that

$$a(u_{N_{\text{max}}, M}^K(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_{N_{\text{max}}, M}^{hp}; \quad (5.9)$$

finally evaluate the hp -DRB output approximation

$$s_{N_{\text{max}}, M}^K(\mu) = \ell(u_{N_{\text{max}}, M}^K(\mu)). \quad (5.10)$$

We define the hp -DRB error bound as

$$\Delta_{N_{\text{max}}, M}^K(\mu) = \frac{\|\mathcal{R}_{N_{\text{max}}, M}^K(\mu)\|_X}{\alpha_{\text{LB}}(\mu)}, \quad (5.11)$$

where $\mathcal{R}_{N_{\text{max}}, M}^K(\mu) \in X$ denotes the Riesz representation of the residual

$$r_{N_{\text{max}}, M}^K(\cdot; \mu) = a(u_{N_{\text{max}}, M}^K(\mu), \cdot; \mu) - f(\cdot; \mu) \in X'. \quad (5.12)$$

We may readily show that $\|u(\mu) - u_{N_{\text{max}}, M}^K(\mu)\|_X \leq \Delta_{N_{\text{max}}, M}^K(\mu)$ by arguments analogous to (3.10)–(3.12). We recall from our discussion in Section 3.4.2 that we may evaluate $\Delta_{N_{\text{max}}, M}^K(\mu)$ inexpensively at cost independent of \mathcal{N} and N .

We emphasize that with these modifications we access entities of truth complexity \mathcal{N} only for the construction of the intermediate RB model (of complexity $N \ll \mathcal{N}$) upon which the hp -DRB approximation is constructed. We discuss the offline-online decoupling of the hp -DRB method in the next subsection.

5.3 Offline–Online Decomposition

The hp -DRB *offline* stage comprises intermediate RB model construction and then hp -DRB partition and approximation space construction based on this underlying RB model.

1. RB model construction. We construct an intermediate RB model over \mathcal{D} : we perform Greedy^{RB} (Algorithm 1) for specified $\mu_1 \in \mathcal{D}$ and $\epsilon_{\text{tol}}^{\text{RB}}$. The cost is \mathcal{N} -dependent.
2. hp -DRB partition and approximation space construction. We construct an hp -DRB model based on the intermediate RB model in step 1 as discussed in the previous two subsections. This step includes, for each DRB approximation space, construction of the parameter-independent entities required for DRB output and DRB error bound evaluation. The cost is \mathcal{N} -independent.

The offline stage may be expensive; however with the DRB modification in step 2 above we significantly reduce the offline computational cost compared to a standard hp -RB approach: the $N_{\text{total}} = \sum_{k=1}^K N^k$ truth FE snapshots of \mathcal{N} -dependent complexity required by the standard hp -RB offline stage are replaced by $M_{\text{total}} = \sum_{k=1}^K M^k$ RB snapshots of N_{max} -dependent complexity.⁶

In the online stage, given any new parameter value $\mu \in \mathcal{D}$, we first determine which subdomain $\mathcal{V}^{k^*} \subset \mathcal{D}$ contains μ through a binary search at cost $\mathcal{O}(\log K)$. Then, for given $1 \leq M \leq M_{\text{max}}^{hp}$, we compute the DRB solution, DRB output, and DRB error bound at cost $\mathcal{O}(M^3 + M^2 Q^2)$. We note that the online cost is independent of the truth complexity \mathcal{N} and the complexity N associated with the underlying intermediate RB model. We emphasize that in the online stage, we invoke the DRB error bound with respect to the FE truth approximation.

We finally note that the offline–online decomposition associated with the hp -DRB approximation is rather different from the offline–online decomposition associated with focus calculations: the DRB “technology” is invoked in the offline (and not online) stage.

⁶Note that we expect here that $M^k \approx N^k$ as long as M^k is significantly smaller than N_{max} , $1 \leq k \leq K$. Also note that for simplicity in this argument we assume that K is the same for the hp -RB and hp -DRB approaches.

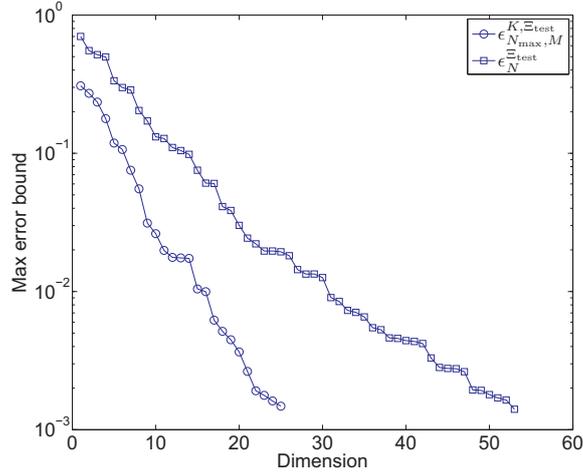


Figure 8: Maximum RB (squares) and hp -DRB (circles) error bounds over random test parameter values as a function of approximation space dimension.

5.4 Numerical Results

5.4.1 Thermal Block

We now apply the hp -DRB method to the thermal block problem introduced in Section 2.2.1. For the underlying intermediate RB space $X_{N_{max}}$ we use the same space as for the thermal block focus calculation example: $N_{max} = 96$. We then pursue the hp -DRB procedure discussed above for $\epsilon_{tol}^h = 0.3$, $\epsilon_{tol}^p = 10^{-3}$, and $M_{max}^{hp} = 25$; the initial parameter value for the partition procedure is $\mu_1^{\mathcal{D}} = (0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5)$. The hp -DRB offline computation results in a partition of \mathcal{D} into $K = 7565$ subdomains, each of which has an associated DRB approximation space of dimension at most $M_{max}^{hp} = 25$.

We now introduce a uniformly distributed random test set $\Xi_{test} \subset \mathcal{D}$ of size $|\Xi_{test}| = 1000$. We then define, for $1 \leq N \leq N_{max}$, the maximum error bound associated with the RB approximation,

$$\epsilon_N^{\Xi_{test}} = \max_{\mu \in \Xi_{test}} \Delta_N(\mu); \quad (5.13)$$

we also define, for $1 \leq M \leq M_{max}^{hp}$, the maximum error bound associated with

the hp -DRB approximation,

$$\epsilon_{N_{\max}, M}^{K, \Xi_{\text{test}}} = \max_{\mu \in \Xi_{\text{test}}} \Delta_{N_{\max}, M}^K(\mu). \quad (5.14)$$

In Figure 8 we compare $\epsilon_N^{\Xi_{\text{test}}}$ and $\epsilon_{N_{\max}, M}^{K, \Xi_{\text{test}}}$ as functions of the approximation space dimensions N and M , respectively: clearly the hp -DRB approximation provides significant dimension reduction. For example, $N = 30$ and $M = 15$ basis functions are required for an error bound of approximately 10^{-2} for the RB and hp -DRB approximation, respectively. The hp -DRB thus provide in this case online computational savings by a factor of 8 (provided the dense system matrix LU-factorization dominates online cost).

The main point of this example is not the dimension reduction provided by the hp -DRB procedure *per se*: we would have obtained similar dimension reduction were we to use a standard hp -RB procedure. Our emphasis here is on the offline stage, which requires 232608 snapshots: this task is feasible in the hp -DRB case in which each snapshot is an RB calculation ($N_{\max} = 96$ degrees of freedom), but would clearly be prohibitive in the standard hp -RB case in which each snapshot is a *truth* calculation ($\mathcal{N} = 9261$ degrees of freedom).

5.4.2 Acoustic Horn

We now apply the hp -DRB method to the acoustic horn problem introduced in Section 2.2.2. For the underlying intermediate RB space $X_{N_{\max}}$ we use the same space as for the acoustic horn focus calculation example: $N_{\max} = 109$. We then pursue the hp -DRB procedure discussed above for $\epsilon_{\text{tol}}^h = 10$, $\epsilon_{\text{tol}}^p = 10^{-4}$, and $M_{\max}^{hp} = 30$; the initial parameter value for the partition procedure is $\mu_1^{\mathcal{D}} = (1.4, 2.15, 1.0)$. The hp -DRB offline computation results in a partition of \mathcal{D} into $K = 997$ subdomains as shown in Figure 9, each of which has an associated DRB approximation space of dimension at most $M_{\max}^{hp} = 30$.

We now introduce a uniformly distributed random test set $\Xi_{\text{test}} \subset \mathcal{D}$ of size $|\Xi_{\text{test}}| = 1000$ and show in Figure 10 $\epsilon_N^{\Xi_{\text{test}}}$ and $\epsilon_{N_{\max}, M}^{K, \Xi_{\text{test}}}$ as functions of the approximation space dimensions N and M , respectively: clearly the hp -DRB approximation provides significant dimension reduction. As for the thermal block example, our main point here is that the DRB strategy enables a feasible hp -DRB offline computation, compared to a prohibitive or infeasible hp -RB offline computation.

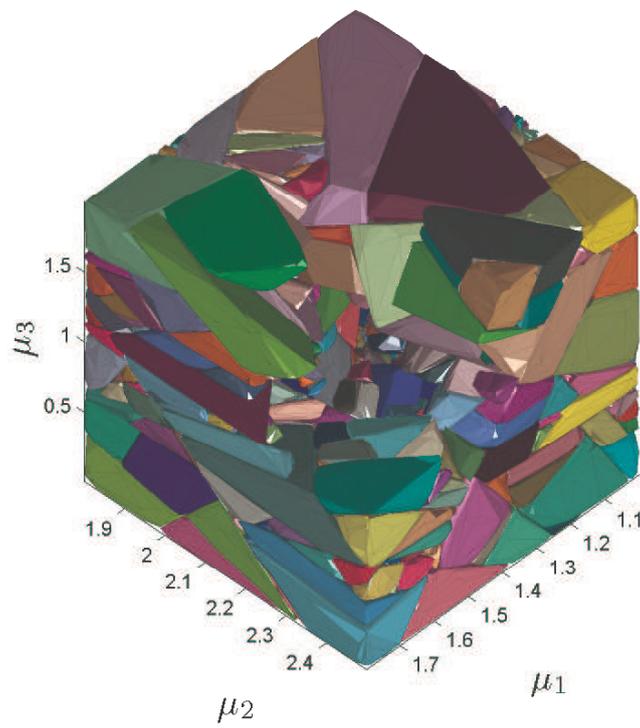


Figure 9: The parameter domain partition associated with the hp -DRB approximation for the acoustic horn problem; note that one octant of the parameter domain is hidden.

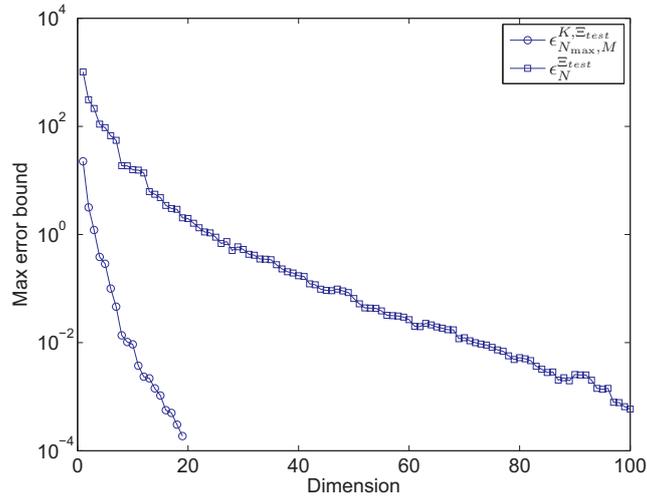


Figure 10: Maximum RB (squares) and hp -DRB (circles) error bounds over random test parameter values as a function of approximation space dimension.

6 Conclusions and Future Work

We have demonstrated that the new DRB method may provide significant on-line speedup in the context of focus calculations, for example for visualization or optimization of RB outputs and RB error bounds over a subdomain or submanifold of the original parameter domain. Further, we have demonstrated that the DRB method may provide significant offline speedup for hp -RB computations, or indeed enable hp -RB computations for problems for which the cost of the standard hp -RB offline stage is prohibitive.

There are several opportunities for extensions. First, the DRB method readily extends to linear parabolic (coercive or non-coercive) problems; we refer to [10, 12] and [6] for (standard) RB and hp -RB treatment of this class of problems, respectively. We may also straightforwardly apply the DRB approach to quadratically nonlinear problems; see [6] for hp -RB treatment of the unsteady incompressible Navier-Stokes equations. Second, we believe that the DRB method will further increase the efficacy of the RB method in applications on “lightweight” hardware [15] where it is crucial to minimize the cost of a reduced order model both in terms of computation time and memory footprint.

In future work we plan to investigate applications of DRB technology in a range of new areas such as *in situ* parameter estimation, uncertainty quantification and design/optimization.

Acknowledgements

This work has been supported by AFOSR Grant number FA9550-07-1-0425, OSD/AFOSR Grant number FA9550-09-1-0613, and the Norwegian University of Science and Technology.

Bibliography

- [1] B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16:525–528, 1978.
- [2] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [3] S. Boyaval. Reduced-basis approach for homogenization beyond the periodic setting. *Multiscale Modeling & Simulation*, 7(1):466–494, 2008.
- [4] S. Boyaval, C. Le Bris, Y. Maday, N. C. Nguyen, and A. T. Patera. A reduced basis approach for variational problems with stochastic parameters: Application to heat conduction with variable robin coefficient. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44):3187 – 3206, 2009.
- [5] C. Canuto, T. Tonn, and K. Urban. A posteriori error analysis of the reduced basis method for nonaffine parametrized nonlinear PDEs. *SIAM Journal on Numerical Analysis*, 47(3):2001–2022, 2009.
- [6] J. L. Eftang, D. J. Knezevic, and A. T. Patera. An *hp* certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems, to appear*, 2011. <http://augustine.mit.edu>.

Paper 4

- [7] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An “*hp*” certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200, 2010.
- [8] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.
- [9] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [10] M. A. Grepl and A. T. Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005.
- [11] B. Haasdonk, M. Dihlmann, and M. Ohlberger. A Training Set and Multiple Bases Generation Approach for Parametrized Model Reduction Based on Adaptive Grids in Parameter Space. Technical Report 28, SRC SimTech, 2010.
- [12] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
- [13] B. Haasdonk and M. Ohlberger. Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Mathematical and Computer Modelling of Dynamical Systems*, 2011. 10.1080/13873954.2010.514703.
- [14] D. B. P. Huynh, D. J. Knezevic, and A. T. Patera. Certified reduced basis model characterization: a frequentistic uncertainty framework. *Computer Methods in Applied Mechanics and Engineering*, submitted, January 2011. <http://augustine.mit.edu>.
- [15] D. B. P. Huynh, D. J. Knezevic, J. W. Peterson, and A. T. Patera. High-fidelity real-time simulation on deployed platforms. *Computers & Fluids*, 43(1):74 – 81, 2011. Symposium on High Accuracy Flow Simulations. Special Issue Dedicated to Prof. Michel Deville, Symposium on High Accuracy Flow Simulations.

- [16] D. B. P. Huynh, D.J. Knezevic, Y. Chen, J.S. Hesthaven, and A.T. Patera. A natural-norm successive constraint method for inf-sup lower bounds. *Computer Methods in Applied Mechanics and Engineering*, 199(29-32):1963 – 1975, 2010.
- [17] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. *Comptes Rendus Mathematique*, 345(8):473 – 478, 2007.
- [18] M. Kamon, F. Wang, and J. White. Generating nearly optimally compact models from Krylov-subspace based reduced-order models. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 47(4):239 –248, apr. 2000.
- [19] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey. libMesh: A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations. *Engineering with Computers*, 22(3–4):237–254, 2006.
- [20] D. J. Knezevic and A. T. Patera. A certified reduced basis method for the fokker–planck equation of dilute polymeric fluids: Fene dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793–817, 2010.
- [21] D. J. Knezevic and J. W. Peterson. A high-performance parallel implementation of the certified reduced basis method. *Computer Methods in Applied Mechanics and Engineering (submitted)*, 2010.
- [22] B. Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control*, 26(1):17 – 32, February 1981.
- [23] N. C. Nguyen, G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for parametrized parabolic pdes: Application to real-time bayesian parameter estimation. In Biegler, Biroso, Ghattas, Heinkenschloss, Keyes, Mallick, Marzouk, Tenorio, Waanders, and Willcox, editors, *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 151–177. John Wiley & Sons, Ltd, 2010.
- [24] A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18:455–462, 1980.

Paper 4

- [25] T. A. Porsching. Estimation of the error in the reduced basis method solution of nonlinear equations. *Mathematics of Computation*, 45(172):487–496, 1985.
- [26] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.
- [27] R. Udawalpola and M. Berggren. Optimization of an acoustic horn with respect to efficiency and directivity. *International Journal for Numerical Methods in Engineering*, 73:1571–1606, 2008.
- [28] K. Veroy, C. Prud’homme, D. V. Rovas, and A. T. Patera. 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.
- [29] K. Willcox and A. Megretski. Fourier series for accurate, stable, reduced-order models in large-scale linear applications. *SIAM Journal on Scientific Computing*, 26(3):944–962 (electronic), 2005.

PAPER 5

PARAMETER MULTI-DOMAIN “*hp*” EMPIRICAL
INTERPOLATION

JENS L. EFTANG AND BENJAMIN STAMM

NTNU Preprint Numerics No. 3/2011

*Submitted to
International Journal for Numerical Methods in Engineering,
2011*

PARAMETER MULTI-DOMAIN “*hp*” EMPIRICAL INTERPOLATION

JENS L. EFTANG* AND BENJAMIN STAMM^{†,‡}

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

*†Department of Mathematics,
University of California, Berkeley,
Berkeley, California, USA*

*‡Lawrence Berkeley National Laboratory,
Berkeley, California, USA*

Abstract

In this paper, we introduce two parameter multi-domain “*hp*” techniques for the empirical interpolation method (EIM). In both approaches, we construct a partition of the original parameter domain into parameter subdomains: *h*-refinement. We apply the standard EIM independently within each subdomain to yield local (in parameter) approximation spaces: *p*-refinement. Further, for a particularly simple case we introduce *a priori* convergence theory for the partition procedure. We show through two numerical examples that our approaches provide significant reduction in the EIM approximation space dimension, and thus significantly reduce the computational cost associated with EIM approximations.

1 Introduction

The Empirical Interpolation Method (EIM) was first introduced in [1, 7] as a tool within the Reduced Basis (RB) framework [12] for parametrized partial differential equations (PDEs). The EIM serves to construct “affine” (more precisely, affine in functions of the parameter) approximations of non-affine parametrized differential operators. This approximation is achieved through an affine approximation of the coefficient function which separates the parameter and spatial dependence. An affine decomposition of the differential operator is necessary to enable efficient RB offline-online computational procedures. The EIM thus expands the class of PDEs amenable to RB treatment; other applications of the

EIM include rapid numerical approximation of parametrized integrals and are discussed in [9].

Given “any” parametrized function, the EIM precomputation (henceforth *offline*) stage serves to construct an approximation space spanned by “snapshots” of this function for judiciously chosen parameter values from a predefined parameter domain *and* a set of judiciously chosen spatial interpolation nodes from the spatial domain. In the EIM online stage, given *any* new parameter value from the parameter domain, the EIM approximation to the original function is the particular linear combination of the EIM basis functions that interpolates the original function at the spatial interpolation nodes.

Under the assumption that the function under consideration depends sufficiently smoothly on the parameters, the EIM typically provides exponential convergence [9]. However, for many problems in which the function exhibits large (albeit smooth) variations with the parameters, a snapshot from one region of the parameter domain contributes little to the approximation of the function associated with a parameter value from another region of the parameter domain. The global (in parameter, but of course also space) EIM approximation space is thus in some sense unnecessarily large, and consequently the online computation of the EIM approximation is unnecessarily expensive.

In this paper, we introduce two approaches that both serve to reduce the dimension of the EIM approximation space. Both approaches share the same underlying idea: an adaptive partition of the parameter domain into parameter subdomains — *h*-refinement — and construction of standard EIM approximation spaces and associated EIM interpolation nodes restricted to each of these parameter subdomains — *p*-refinement. This parameter multi-domain, or *hp*, strategy provides significant dimension reduction since the smaller local (in parameter) EIM approximation spaces are optimized with respect to the parametric variations within each subdomain; the online evaluation of the EIM interpolant is thus much faster.

Our first approach is the *anchor point* (AP) splitting scheme. This method is an adaption of the *hp*-RB method introduced in [4] to the context of the EIM. The parameter subdomains are hierarchically defined based on proximity to “anchor points” identified by the EIM Greedy sampling procedure within each subdomain at each level of *h*-refinement; subsequently, in the *p*-refinement stage, the standard EIM is applied within each subdomain. Our second approach is the *gravity center* (GC) splitting scheme. The parameter subdomains are hierarchically defined based on the “gravity center” of a cloud of points identified by the EIM Greedy sampling procedure at each level of concurrent *h*- and *p*-refinement.

We provide in the next section the problem statement along with notation required later. We then review in Section 3 the standard EIM applied to the entire parameter domain \mathcal{D} . Then, we present in Section 4 and Section 5 the AP and GC splitting procedures, respectively. In Section 6 we discuss the computational cost associated with both methods. In Section 7 we compare our two approaches relative to the standard EIM through two model problems. Finally, we provide some concluding remarks in Section 8.

2 Problem statement

We introduce a spatial domain $\Omega \subset \mathbb{R}^d$ for some integer $d > 0$; we shall denote a particular spatial point $x \in \Omega$ as $x = (x_{(1)}, \dots, x_{(d)})$. We next introduce a parameter domain $\mathcal{D} \subset \mathbb{R}^P$; we shall denote a particular parameter value $\mu \in \mathcal{D}$ as $(\mu_{(1)}, \dots, \mu_{(P)})$. We then introduce a (given) parametrized function $\mathcal{G} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ such that $\mathcal{G}(\cdot; \mu) \in L^\infty(\Omega)$ for all $\mu \in \mathcal{D}$; here $L^\infty(\Omega) = \{v : \text{ess sup}_{x \in \Omega} |v(x)| < \infty\}$. We finally introduce a triangulation $\mathcal{T}^{\mathcal{N}}(\Omega)$ with \mathcal{N} vertices over which we shall in practice realize $\mathcal{G}(\cdot; \mu)$, $\mu \in \mathcal{D}$, as a piecewise linear function.¹

For any $\mu \in \mathcal{D}$, we consider the construction of an approximation $\mathcal{G}_M(\cdot; \mu) \approx \mathcal{G}(\cdot; \mu)$, where $\mathcal{G}_M(\cdot; \mu)$ resides in a parameter-independent M -dimensional linear approximation space W_M , $M < \infty$. The problem is thus twofold: *i*) the construction of a good M -dimensional approximation space $W_M = \text{span}\{q_1, \dots, q_M\}$ and *ii*) given any $\mu \in \mathcal{D}$ and the space W_M , the computation of parameter dependent coefficients $\varphi_1(\mu), \dots, \varphi_M(\mu)$ such that

$$\mathcal{G}_M(\cdot; \mu) = \sum_{i=1}^M \varphi_i(\mu) q_i \approx \mathcal{G}(\cdot; \mu) \quad (2.1)$$

is a good approximation. Clearly, classical polynomial interpolation procedures may be considered for this problem; however in our context here standard polynomial approximation spaces are far too general and hence the required dimension M is too large to accommodate efficient online evaluation of $\varphi_1(\mu), \dots, \varphi_M(\mu)$. In contrast, the EIM provides a much smaller approximation space specifically targeted at the parametrized function at hand.

¹We emphasize that the EIM is not restricted to functions that are piecewise linear; however, for the computational procedures involved, a finite-dimensional representation of $\mathcal{G}(\cdot; \mu)$, $\mu \in \mathcal{D}$, is required.

The following simple problem illustrates how the EIM may be invoked in practice. Consider the integral

$$F(\mu) = \int_{\Omega} f(\cdot; \mu)u, \quad (2.2)$$

where $f : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ with $f(\cdot; \mu) \in L^\infty(\Omega)$ for any $\mu \in \mathcal{D}$ and $u : \Omega \rightarrow \mathbb{R}$ is a parameter independent function (we assume that the product $f(\cdot; \mu)u$ is integrable). In general, evaluation of $F(\mu)$ with standard quadrature rules may be expensive; in particular, the evaluation cost may be prohibitive when $F(\mu)$ has to be computed for many $\mu \in \mathcal{D}$ or in real time. The EIM serves to construct an approximation $f_M(\cdot; \mu) = \sum_{m=1}^M \varphi_m(\mu)q_m$ to $f(\cdot; \mu)$ such that

$$F_M(\mu) \equiv \int_{\Omega} f_M(\cdot; \mu)u = \sum_{m=1}^M \varphi_m(\mu) \int_{\Omega} q_m u \quad (2.3)$$

is a good approximation to $F(\mu)$. The key point is that the separation provided by the EIM allows precomputation of the integrals in (2.3) (by for example standard quadrature rules). Hence subsequent evaluation $\mu \rightarrow F_M(\mu)$ may be performed very fast.

3 The empirical interpolation method

The EIM was originally proposed in [1] (see also [7] for a more elaborate presentation). In this paper however we shall employ the particular version of the EIM introduced in [9], which invokes the less expensive interpolation error rather than the more expensive projection error as a tool in the offline construction of the EIM approximation space. We now briefly review the EIM applied to the entire parameter domain \mathcal{D} ; in the next sections we then consider the EIM within the hp context.

We first introduce the empirical interpolation of a function $\mathcal{G} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$; we require that $\mathcal{G}(\cdot; \mu) \in L^\infty(\Omega)$ for all $\mu \in \mathcal{D}$. We introduce the EIM space $W_M = \text{span}\{q_m\}_{m=1}^M$ of dimension M and the M EIM interpolation nodes $t_1, \dots, t_M \in \Omega$ (the EIM basis functions q_m , $1 \leq m \leq M$, and interpolation nodes will be defined shortly). We may now define, for any $\mu \in \mathcal{D}$, the empirical interpolation $\mathcal{G}_M(\cdot; \mu) \approx \mathcal{G}(\cdot; \mu)$, as the particular function in W_M that interpolates $\mathcal{G}(\cdot; \mu)$ at

the M interpolation nodes:

$$\mathcal{G}_M(\cdot; \mu) = \sum_{i=1}^M \varphi_i(\mu) q_i, \quad (3.1)$$

where the coefficients $\varphi_i(\mu)$, $1 \leq i \leq M$, solve the linear system

$$\sum_{j=1}^M \varphi_j(\mu) q_j(t_i) = \mathcal{G}(t_i; \mu), \quad 1 \leq i \leq M. \quad (3.2)$$

It is easy to see that $\mathcal{G}_M(t_i; \mu) = \mathcal{G}(t_i; \mu)$, $1 \leq i \leq M$, for all $\mu \in \mathcal{D}$.

We now define the EIM basis functions and the EIM interpolation nodes recursively through a Greedy sampling algorithm. To this end we require a (typically rich) training set $\Xi_{\text{train}} \subset \mathcal{D}$ of finite size $|\Xi_{\text{train}}|$ which shall serve as our computational surrogate for \mathcal{D} . First, for $M = 1$, we choose (randomly, say) an initial parameter value $\mu_1 \in \mathcal{D}$; the first EIM interpolation node is then $t_1 \equiv \arg \sup_{x \in \Omega} |\mathcal{G}(x; \mu_1)|$; the first EIM basis function is $q_1 \equiv \mathcal{G}(\cdot; \mu_1) / \mathcal{G}(t_1; \mu_1)$. Then, for $2 \leq M \leq M_{\text{max}} < \infty$, we compute for all $\mu \in \Xi_{\text{train}} \subset \mathcal{D}$ the empirical interpolation $\mathcal{G}_{M-1}(\cdot; \mu) \approx \mathcal{G}(\cdot; \mu)$; the next parameter is then chosen as the maximizer of the $L^\infty(\Omega)$ interpolation error over Ξ_{train} :

$$\mu_M \equiv \arg \max_{\mu \in \Xi_{\text{train}}} \|\mathcal{G}_{M-1}(\cdot; \mu) - \mathcal{G}(\cdot; \mu)\|_{L^\infty}. \quad (3.3)$$

We define $r_M \equiv \mathcal{G}_{M-1}(\cdot; \mu_M) - \mathcal{G}(\cdot; \mu_M)$ and choose the next EIM interpolation node as

$$t_M \equiv \arg \sup_{x \in \Omega} |r_M(x)|. \quad (3.4)$$

We may now finally define the next EIM basis function as

$$q_M \equiv \frac{r_M}{r_M(t_M)}. \quad (3.5)$$

We have thus obtained M_{max} basis functions and M_{max} interpolation nodes. We note that by construction $r_M(t_i) = 0$ for $1 \leq i \leq M-1$; hence $q_M(t_i) = 0$ for $1 \leq i \leq M-1$ and $q_M(t_M) = 1$ thanks to the normalization (3.5). The matrix

²Note that $\sup_{x \in \Omega} |\mathcal{G}(x; \mu)|$ is in practice realized as the maximum of $|\mathcal{G}(x; \mu)|$ over the \mathcal{N} vertices of $\mathcal{T}^{\mathcal{N}}(\Omega)$.

$\{q_j(t_i)\}_{i,j}$ in (3.2) is thus lower triangular with unity diagonal; as a result, for any $\mu \in \mathcal{D}$, the cost associated with the computation of the coefficients $\varphi_j(\mu)$, $1 \leq j \leq M$, is $\mathcal{O}(M^2)$.

For $1 \leq M \leq M_{\max}$, we define the ‘‘Lebesgue constant’’ [10]

$$\Lambda_M \equiv \sup_{x \in \Omega} \sum_{m=1}^M |V_m^M(x)|, \quad (3.6)$$

where $V_m^M \in W_M$, $1 \leq m \leq M$, are the ‘‘characteristic functions’’ of W_M , which satisfy $V_m^M(t_n) = \delta_{mn}$; here δ_{mn} is the Kronecker delta symbol. It can be proven [1, 7] that the EIM approximation error satisfies

$$\|\mathcal{G}(\cdot; \mu) - \mathcal{G}_M(\cdot; \mu)\|_{L^\infty} \leq (1 + \Lambda_M) \inf_{z \in W_M} \|\mathcal{G}(\cdot; \mu) - z\|_{L^\infty}, \quad 1 \leq M \leq M_{\max}, \quad (3.7)$$

Furthermore, it can be proven that $\Lambda_M \leq 2^M - 1$; however in actual practice the behavior of Λ_M is much better [1, 7, 9].

4 An anchor point splitting scheme

4.1 Procedure

In this section we introduce the anchor point (AP) splitting procedure for the partition of the parameter domain; this procedure is an adaption of the approach introduced for the *hp* reduced basis method in [3, 4]. We shall require a *distance function* $d : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$, which we choose in this paper as the Euclidean distance between the two arguments; however, other distance functions may be considered.

We first describe the splitting of an arbitrary subdomain $\mathcal{V} \subseteq \mathcal{D}$ into two distinct subdomains $\mathcal{V}_0 \subset \mathcal{V}$ and $\mathcal{V}_1 \subset \mathcal{V}$; the application of this splitting step to the construction of a partition of \mathcal{D} is straightforward and is discussed shortly. We assume that \mathcal{V} is equipped with a sufficiently dense training set $\Xi_{\text{train}}^\mathcal{V} \subset \mathcal{V}$. Given an *anchor point* $\mu_0^* \in \mathcal{V}$, we set $\mu_1 = \mu_0^*$; we then compute $\mathcal{G}(\cdot; \mu_1)$, $t_1 = \arg \sup_{x \in \Omega} |\mathcal{G}(x; \mu_1)|$, and perform one iteration of the standard EIM Greedy procedure restricted to $\Xi_{\text{train}}^\mathcal{V}$ (hence $M_{\max} = 2$). We then define $\mu_1^* \equiv \mu_2$ and we denote the maximum interpolation error over $\Xi_{\text{train}}^\mathcal{V}$ by

$$\epsilon^\mathcal{V} \equiv \max_{\mu \in \Xi_{\text{train}}^\mathcal{V}} \|\mathcal{G}_1(\cdot; \mu) - \mathcal{G}(\cdot; \mu)\|_{L^\infty}. \quad (4.1)$$

We can now define two distinct subdomains $\mathcal{V}_0 \subset \mathcal{V}$ and $\mathcal{V}_1 \subset \mathcal{V}$ based on proximity to the two points μ_0^* and μ_1^* as

$$\mathcal{V}_0 = \{\mu \in \mathcal{V} : d(\mu, \mu_0^*) < d(\mu, \mu_1^*)\}, \quad (4.2)$$

$$\mathcal{V}_1 = \{\mu \in \mathcal{V} : d(\mu, \mu_1^*) \leq d(\mu, \mu_0^*)\}. \quad (4.3)$$

We say that μ_0^* is to the anchor point of \mathcal{V}_0 and that μ_1^* is the anchor point of \mathcal{V}_1 .

We apply this “ h -refinement” splitting scheme in a recursive manner in order to construct a hierarchical partition of the entire domain \mathcal{D} : we first choose the initial anchor point—typically a corner of \mathcal{D} —and split \mathcal{D} into two new subdomains. We then apply the splitting scheme within each of the two generated subdomains. We continue recursively until convergence: we split a subdomain \mathcal{V} as long as the maximum error $\epsilon^{\mathcal{V}}$ in (4.1) is larger than a prescribed tolerance $\epsilon_{\text{tol}}^h > 0$. If $\epsilon^{\mathcal{V}} < \epsilon_{\text{tol}}^h$ we stop the splitting process. We note that each subdomain (except the “root” \mathcal{D}) has a single “parent” and one “sibling.” Thanks to this structure we may organize the splitting procedure in a binary tree as illustrated in Figure 1.

When the tolerance ϵ_{tol}^h is satisfied over $\Xi_{\text{train}}^{\mathcal{V}}$, we perform “ p -refinement” within \mathcal{V} : application of the standard EIM to \mathcal{V} for specified $M_{\text{max}} > 1$ and target tolerance $\epsilon_{\text{tol}}^p < \epsilon_{\text{tol}}^h$. If the target tolerance ϵ_{tol}^p is *not* satisfied after p -refinement (for M_{max} basis functions), we successively perform additional “ h ” and “ p ” refinement steps until the tolerance is satisfied for at most M_{max} EIM basis functions. Our procedure thus enables simultaneous control over the EIM error (over the training set) and the EIM space dimension (and thus online cost).

This hp -EIM anchor point refinement procedure results in a finite number K of parameter subdomains, which we label $\mathcal{V}_1, \dots, \mathcal{V}_K$. Each of these subdomains has an associated nested set of EIM approximation spaces, $W_M^k = \text{span}\{q_m^k\}_{m=1}^M$, $1 \leq M \leq M_{\text{max}}^k$, $1 \leq k \leq K$, and an associated set of nested EIM interpolation nodes $T_M^k = \{t_1^k, \dots, t_M^k\}$, $1 \leq M \leq M_{\text{max}}^k$, $1 \leq k \leq K$. Here, the q_m^k denote the EIM basis functions, and M_{max}^k denotes the space dimension required in order to reach the target tolerance for subdomain k . Note that the M_{max}^k , $1 \leq k \leq K$, are in general different but bounded by M_{max} .

Given the partition of \mathcal{D} into K subdomains with associated EIM approximation spaces and interpolation nodes, we now define the AP hp -EIM interpolant. Given any $\mu \in \mathcal{D}$, we first determine $k^* = k^*(\mu)$ such that $\mu \in \mathcal{V}_{k^*}$. Note that thanks to the binary tree structure of the partition, determination of k^* is an

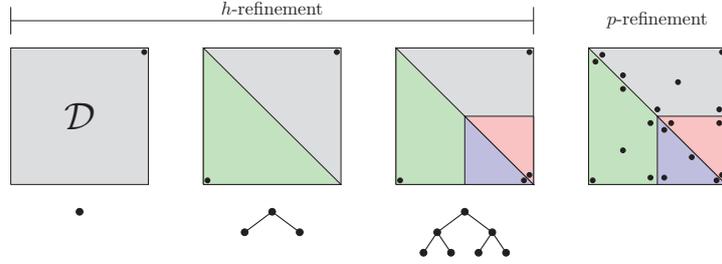


Figure 1: Two levels of h -refinement and subsequent p -refinement for the anchor point splitting procedure.

efficient binary search. We then compute the EIM approximation as

$$\mathcal{G}_{M,K}(\cdot; \mu) \equiv \mathcal{G}_{M,K}^{k^*}(\cdot; \mu) = \sum_{i=1}^M \varphi_{M,i}^{k^*}(\mu) q_i^{k^*}, \quad (4.4)$$

where the coefficients $\varphi_{M,i}^{k^*}(\mu)$, $1 \leq i \leq M$, solve the linear system

$$\sum_{j=1}^M \varphi_{M,j}^{k^*}(\mu) q_j^{k^*}(t_i^{k^*}) = \mathcal{G}(t_i^{k^*}; \mu), \quad 1 \leq i \leq M, \quad (4.5)$$

Remark 1. We note that the partition of \mathcal{D} can be organized in a binary tree as indicated in Figure 1 regardless of the parameter dimension P , since we always subdivide a subdomain into two new subdomains at each level of refinement. The method thus allows the partition to reflect anisotropy in the underlying parameter dependence of $\mathcal{G}(\cdot; \mu)$.

4.2 An *a priori* convergence theory for the AP procedure

We present here an *a priori* theory for the convergence of the initial partition algorithm (h -refinement) presented above. The theory does not consider the subsequent p -refinement however this step will ultimately (trivially) converge since the training sets are of finite size. Our interest is not in the asymptotic convergence of the partition procedure *per se*, since in practice we will always invoke an hp -type approximation rather than a pure h -type approximation. However, the theory suggests that our procedure generates a meaningful

partition, and furthermore guarantees that the partition procedure does in fact terminate for specified ϵ_{tol}^h . We consider the case with $P = 1$ parameter for simplicity.

Proposition 1. *Let $\mathcal{D} \subset \mathbb{R}$ and let $|\mathcal{D}|$ denote the length of \mathcal{D} . Suppose that $\mathcal{G}(\cdot; \mu)$ is Lipschitz-continuous in μ with Lipschitz constant $L < \infty$: for any $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$,*

$$\|\mathcal{G}(\cdot; \mu_1) - \mathcal{G}(\cdot; \mu_2)\|_{L^\infty} \leq L|\mu_1 - \mu_2|. \quad (4.6)$$

For any specified $\epsilon_{\text{tol}}^h > 0$, the AP splitting procedure is then convergent for $K = K(\epsilon_{\text{tol}}^h)$ subdomains; moreover, the convergence is first order in the sense that

$$K(\epsilon_{\text{tol}}^h) \leq \max\left\{1, \frac{4L|\mathcal{D}|}{\epsilon_{\text{tol}}^h}\right\}, \quad (4.7)$$

where $|\mathcal{D}|$ is the length of \mathcal{D} .

Proof. We consider our splitting procedure after generation of \tilde{K} subdomains. Either we obtain convergence for $\tilde{K} = 1$ (i.e., $K = 1$) — in which case the proof is complete — or $\tilde{K} > 1$. We henceforth consider the case $\tilde{K} > 1$.

We consider the splitting of an arbitrary subdomain $\mathcal{V} \subset \mathcal{D}$ into distinct subdomains $\mathcal{V}_0 \subset \mathcal{V}$ and $\mathcal{V}_1 \subset \mathcal{V}$ as discussed in the previous subsection. We assume that the error tolerance ϵ_{tol}^h is not satisfied, hence $\epsilon^\mathcal{V} > \epsilon_{\text{tol}}^h$. Let $\mu_0^* \in \mathcal{V}$ denote the anchor point associated with \mathcal{V} . We then consider the empirical interpolation $\tilde{\mathcal{G}}(\cdot; \mu) = \tilde{\varphi}(\mu)\mathcal{G}(\cdot; \mu_0^*) \approx \mathcal{G}(\cdot; \mu)$ for any $\mu \in \mathcal{V}$. For the error in this approximation we obtain

$$\begin{aligned} \|\mathcal{G}(\cdot; \mu) - \tilde{\mathcal{G}}(\cdot; \mu)\|_{L^\infty} &\leq (1 + \Lambda_{(M=1)}) \inf_{z \in \text{span}\{\mathcal{G}(\cdot; \mu_0^*)\}} \|\mathcal{G}(\cdot; \mu) - z\|_{L^\infty} \\ &\leq 2\|\mathcal{G}(\cdot; \mu) - \mathcal{G}(\cdot; \mu_0^*)\|_{L^\infty} \\ &\leq 2L|\mu - \mu_0^*|, \end{aligned} \quad (4.8)$$

where we first invoke (3.7), then choose $z = \mathcal{G}(\cdot; \mu_0^*)$, and finally invoke (4.6). Note that it follows from the definition of Λ_1 (Eq. (3.6)) and the characteristic function V_1^1 that $\Lambda_1 = 1$.

We now let $\mu = \mu_1^*$ denote the anchor point associated with \mathcal{V}_1 , identified by the single EIM Greedy iteration over \mathcal{V} . Eq. (4.8) then yields

$$\|\mathcal{G}(\cdot; \mu_1^*) - \tilde{\mathcal{G}}(\cdot; \mu_1^*)\|_{L^\infty} \leq 2L|\mu_1^* - \mu_0^*|. \quad (4.9)$$

Since the error tolerance is not satisfied over \mathcal{V} , we have $\epsilon^{\mathcal{V}} = \|\mathcal{G}(\cdot; \mu_1^*) - \tilde{\mathcal{G}}(\cdot; \mu_1^*)\|_{L^\infty} > \epsilon_{\text{tol}}^h$. Hence

$$\epsilon_{\text{tol}}^h < 2L|\mu_1^* - \mu_0^*|. \quad (4.10)$$

We split \mathcal{V} into $\mathcal{V}_0 \subset \mathcal{V}$, $\mathcal{V}_1 \subset \mathcal{V}$ based on Euclidean distance to the two anchor points. It is clear that the length of each subdomain, $|\mathcal{V}_0|$ and $|\mathcal{V}_1|$, is at least as large as half the distance between the anchor points. We thus obtain

$$|\mathcal{V}_i| \geq \frac{|\mu_1^* - \mu_0^*|}{2} > \frac{\epsilon_{\text{tol}}^h}{4L}, \quad i = 0, 1. \quad (4.11)$$

We denote the \tilde{K} subdomains generated by the algorithm so far by $\mathcal{S}_k \subset \mathcal{D}$, $1 \leq k \leq \tilde{K}$; we denote the length of \mathcal{S}_k by $|\mathcal{S}_k|$. Each of these subdomains results from a splitting of a subdomain $\tilde{\mathcal{S}}_k \supset \mathcal{S}_k$ one level further up in the tree. Since \mathcal{V} above was arbitrary, we can for any k , $1 \leq k \leq \tilde{K}$, set $\mathcal{V} = \tilde{\mathcal{S}}_k$ and conclude that

$$|\mathcal{S}_k| > \frac{\epsilon_{\text{tol}}^h}{4L}, \quad 1 \leq k \leq \tilde{K}. \quad (4.12)$$

We define the length of the smallest subdomain as $\delta_{\tilde{K}} \equiv \min_{1 \leq k \leq \tilde{K}} |\mathcal{S}_k|$, and hence in particular $\delta_{\tilde{K}} > \epsilon_{\text{tol}}^h/(4L)$.

We complete the proof by a contradiction argument. Assume that $\tilde{K} > 4L|\mathcal{D}|/\epsilon_{\text{tol}}^h$. In this case

$$\tilde{K}\delta_{\tilde{K}} > \frac{4L|\mathcal{D}|}{\epsilon_{\text{tol}}^h}\delta_{\tilde{K}} > \frac{4L|\mathcal{D}|}{\epsilon_{\text{tol}}^h} \cdot \frac{\epsilon_{\text{tol}}^h}{4L} = |\mathcal{D}|. \quad (4.13)$$

On the other hand, it is clear that $\tilde{K}\delta_{\tilde{K}} \leq |\mathcal{D}|$ for \tilde{K} subdomains. We have thus reached a contradiction, and we conclude that the algorithm can not generate $\tilde{K} > 4L|\mathcal{D}|/\epsilon_{\text{tol}}^h$ subdomains as long as the error tolerance is not satisfied. Hence the error tolerance must be satisfied for, and thus the algorithm must terminate for, $K \leq 4L|\mathcal{D}|/\epsilon_{\text{tol}}^h$ subdomains. \square

5 A gravity center splitting scheme

5.1 Procedure

In this section we introduce the gravity center (GC) splitting procedure. The GC procedure is similar to the AP procedure of the previous section: both

approaches invoke the standard EIM greedy sampling procedure in a recursive way in order to generate hierarchical partitions. However the GC procedure also differs significantly from the AP procedure in several ways: the GC procedure splits a given subdomain in a structured way based on the location of the gravity center of $M_{\max} > 2$ parameter values. Hence the GC splitting procedure also involves higher order approximation terms and, in contrast to the AP splitting, leads to a tensorized partition structure. Below, we shall require an operation $\text{op}(i, j)$ defined as

$$\text{op}(i, j) = \begin{cases} \leq & \text{if } \text{bin}(i-1)_{(j)} = 0 \\ > & \text{if } \text{bin}(i-1)_{(j)} = 1 \end{cases} \quad (5.1)$$

where $\text{bin}(i)$ is the binary representation of i as a vector in $\{0, 1\}^P$.

We first describe the splitting of an “arbitrary” subdomain $\mathcal{V} = [a_{(1)}, b_{(1)}] \times \dots \times [a_{(P)}, b_{(P)}] \subset \mathcal{D}$ into 2^P distinct subdomains $\mathcal{V}_i \subset \mathcal{V}$, $1 \leq i \leq 2^P$, each of which may be written on tensor-product form $\mathcal{V}_i = [a_{(1)}^i, b_{(1)}^i] \times \dots \times [a_{(P)}^i, b_{(P)}^i]$. The application of this splitting step to the construction of a partition of \mathcal{D} is discussed shortly.

First, we perform a standard EIM procedure within \mathcal{V} for a target tolerance ϵ_{tol}^p and a maximum EIM space dimension M_{\max} . If the target tolerance is satisfied over \mathcal{V} , i.e., $\epsilon^{\mathcal{V}} < \epsilon_{\text{tol}}^p$, we terminate the procedure since further splitting of \mathcal{V} domain is not required. If the target tolerance is *not* satisfied over \mathcal{V} , we obtain M_{\max} parameter values $\{\mu_1, \dots, \mu_{M_{\max}}\}$ from the EIM greedy procedure. We then define the gravity point of the point cloud $\{\mu_i\}_{i=1}^{M_{\max}}$ by

$$g = \frac{1}{M_{\max}} \sum_{i=1}^{M_{\max}} \mu_i. \quad (5.2)$$

The i th subdomain \mathcal{V}_i is defined by

$$\mathcal{V}_i = \{\mu \in \mathcal{V} : \mu_{(j)} \text{ op}(i, j) g_{(j)}, \forall 1 \leq j \leq P\}; \quad (5.3)$$

where $\mu_{(j)}$ and $g_{(j)}$ represents the j th element of μ and g , respectively,

As with the AP procedure, we apply the GC splitting scheme recursively in order to construct a hierarchical partition of the original domain \mathcal{D} : we start with a standard EIM greedy procedure within \mathcal{D} , and split \mathcal{D} into 2^P new subdomains. We then apply the GC scheme within each of these subdomains, and continue the procedure recursively until convergence: the target tolerance is achieved over each subdomain with maximum EIM space dimension M_{\max} .

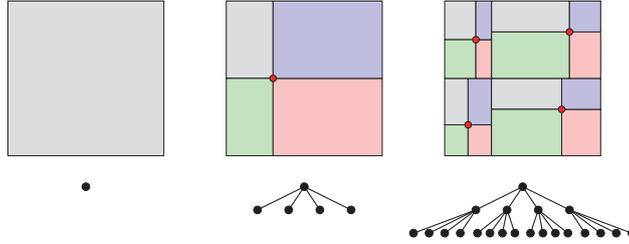


Figure 2: Two levels of parameter domain splitting for the gravity center procedure.

This hp -EIM refinement procedure results in K parameter subdomains, which we label $\mathcal{V}_1, \dots, \mathcal{V}_K$. Each of these subdomains has an associated nested set of EIM approximation spaces, $W_M^k = \text{span}\{q_m^k\}_{m=1}^M$, $1 \leq M \leq M_{\max}^k$, $1 \leq k \leq K$, and an associated set of nested EIM interpolation nodes $T_M^k = \{t_1^k, \dots, t_M^k\}$, $1 \leq M \leq M_{\max}^k$, $1 \leq k \leq K$; as before M_{\max}^k denotes the space dimension required in order to satisfy the target tolerance ϵ_{tol}^p for subdomain k .³

Given the partition of \mathcal{D} into K subdomains with associated EIM approximation spaces and interpolation nodes, we now define the GC hp -EIM approximation. Given any $\mu \in \mathcal{D}$, we first determine $k^* = k^*(\mu)$ such that $\mu \in \mathcal{V}_{k^*}$. Note that thanks to the hierarchical structure of the partition, determination of k^* is an efficient 2^P -order search. We then compute the EIM approximation as

$$\mathcal{G}_{M,K}(\cdot; \mu) \equiv \mathcal{G}_{M,K}^{k^*}(\cdot; \mu) = \sum_{i=1}^M \varphi_{M,i}^{k^*}(\mu) q_i^{k^*}, \quad (5.4)$$

where the coefficients $\varphi_{M,i}^{k^*}(\mu)$, $1 \leq i \leq M$, solve the linear system

$$\sum_{j=1}^M \varphi_{M,j}^{k^*}(\cdot; \mu) q_j^{k^*}(t_i^{k^*}) = \mathcal{G}(t_i^{k^*}; \mu), \quad 1 \leq i \leq M. \quad (5.5)$$

Remark 2. We note that the partition of \mathcal{D} can be organized in a 2^P order tree as indicated in Figure 2, since at each level of refinement we subdivide a

³Strictly speaking, we should here introduce separate notation for the AP and GC splitting procedures. In particular, the number of subdomains K as well as entities associated with each subdomain (such as W_M^k, T_M^k, M_{\max}^k) should bear subscripts AP and GC. However, we omit these subscripts for simplicity of notation. When we later compare the two approaches in terms of numerical results, we introduce separate notation only as necessary.

subdomain into 2^P new subdomains. As a result, the scheme does only take anisotropy in the underlying parameter dependence into account in a weak manner, and as a consequence may construct more subdomains than required. We provide further comments on this issue in Section 7.

5.2 An *a priori* convergence theory for the GC procedure

We present here an *a priori* theory for the convergence of the GC partition algorithm. The theory ensures that the algorithm does in fact terminate for specified ϵ_{tol}^p and M_{max} . We consider the case with $P = 1$ parameter for simplicity.

Proposition 2. *Let $\mathcal{D} \subset \mathbb{R}$ and let $|\mathcal{D}|$ denote the length of \mathcal{D} . Suppose that $\mathcal{G}(\cdot; \mu)$ is Lipschitz-continuous in μ with Lipschitz constant $L < \infty$: for any $\mu_1 \in \mathcal{D}$, $\mu_2 \in \mathcal{D}$,*

$$\|\mathcal{G}(\cdot; \mu_1) - \mathcal{G}(\cdot; \mu_2)\|_{L^\infty} \leq L|\mu_1 - \mu_2|. \quad (5.6)$$

For any specified $\epsilon_{\text{tol}}^p > 0$ and $M_{\text{max}} \geq 2$, the “gravity center” splitting procedure is then convergent for $K = K(\epsilon_{\text{tol}}^p)$ subdomains; moreover, the convergence is first order in the sense that

$$K(\epsilon_{\text{tol}}^h) \leq \max\left\{1, \frac{2(1 + \Lambda_M^{\text{max}})L|\mathcal{D}|}{\epsilon_{\text{tol}}^p}\right\}, \quad (5.7)$$

where $|\mathcal{D}|$ is the length of \mathcal{D} and $\Lambda_M^{\text{max}} \equiv \max_{1 \leq M \leq M_{\text{max}}} \Lambda_M$.

Proof. We demonstrate here only a lower bound for the length of a subdomain (analogously to (4.11)). The remainder of the proof is then identical to the proof of Proposition 1.

We consider the splitting of an arbitrary subdomain $\mathcal{V} = [a, b] \subset \mathcal{D}$ into distinct subdomains \mathcal{V}_1 and \mathcal{V}_2 as discussed in the previous subsection. We assume that the error tolerance ϵ_{tol}^p is not satisfied. In \mathcal{V} we choose by virtue of the standard EIM procedure M_{max} parameter values μ_m , $1 \leq m \leq M_{\text{max}}$, and compute the associated snapshots $\mathcal{G}(\cdot; \mu_m)$, $1 \leq m \leq M_{\text{max}}$. For any $\mu \in \mathcal{V}$ and $1 \leq M \leq M_{\text{max}}$ we consider the empirical interpolation $\tilde{\mathcal{G}}_M(\cdot; \mu) = \sum_{m=1}^M \tilde{\varphi}_m(\mu) \mathcal{G}(\cdot; \mu_m) \approx \mathcal{G}(\cdot; \mu)$. Let $\tilde{W}_M = \text{span}\{\mathcal{G}(\cdot; \mu_m)\}_{m=1}^M$ denote the EIM space associated with the current subdomain. For the EIM approximation error

we obtain, for any $\mu \in \mathcal{V}$,

$$\|\mathcal{G}(\cdot; \mu) - \tilde{\mathcal{G}}_M(\cdot; \mu)\|_{L^\infty} \leq (1 + \Lambda_M) \inf_{z \in \tilde{W}_M} \|\mathcal{G}(\cdot; \mu) - z\|_{L^\infty} \quad (5.8)$$

$$\leq (1 + \Lambda_M) \|\mathcal{G}(\cdot; \mu) - \mathcal{G}(\cdot; \mu_m)\|_{L^\infty} \quad (5.9)$$

$$\leq (1 + \Lambda_M)L|\mu - \mu_m|, \quad (5.10)$$

for $1 \leq m \leq M$ and $1 \leq M \leq M_{\max}$. Since the error tolerance has not been satisfied, we have $\epsilon_{\text{tol}}^p \leq \|\mathcal{G}(\cdot; \mu_{M+1}) - \tilde{\mathcal{G}}_M(\cdot; \mu_{M+1})\|_{L^\infty(\Omega)}$ for $1 \leq M \leq M_{\max}$ (note that here, $\mu_{M_{\max}+1}$ is the parameter value at which the maximum error is obtained with M_{\max} basis functions). With (5.10) we thus obtain

$$|\mu_{M+1} - \mu_m| \geq \frac{\epsilon_{\text{tol}}^p}{(1 + \Lambda_M)L} \quad (5.11)$$

for $1 \leq m \leq M$ and $1 \leq M \leq M_{\max}$.

The subdomain \mathcal{V} is split at its gravity center g defined in (5.2): $\mathcal{V}_1 = [a, g]$ and $\mathcal{V}_2 = (g, b]$. We now bound g away from a and b , and thus obtain a lower bound for $|\mathcal{V}_1| = |g - a|$ and $|\mathcal{V}_2| = |b - g|$. We consider only $|\mathcal{V}_1|$ since the argument for $|\mathcal{V}_2|$ is analogous.

It is clear that there exist i, j , $1 \leq i, j \leq M_{\max}$, such that $\mu_i < g < \mu_j$ since the gravity center must reside inside the convex hull of the $M_{\max} \geq 2$ Greedily chosen parameter values. We consider the situation in which only one of these M_{\max} parameter values resides to the left of g (otherwise $|\mathcal{V}_1| \geq |\mu_{M+1} - \mu_m|$ for some $1 \leq M \leq M_{\max} - 1$ and some $1 \leq m \leq M$). We denote this particular parameter value by μ_i . In this case *at least* one Greedily chosen parameter value μ_j resides to the right of g ; hence $|g - \mu_i| \geq |\mu_i - \mu_j|/2$ since g must be closer to μ_j than to μ_i . Next, assume that μ_i was chosen by the Greedy algorithm at iteration M for $1 \leq M \leq M_{\max} - 1$, and that μ_j was chosen at iteration m for $1 \leq m \leq M - 1$. We thus set $\mu_{M+1} = \mu_i$ and $\mu_m = \mu_j$ in (5.11) (if μ_i was chosen by the Greedy algorithm *before* μ_j , we set $\mu_{M+1} = \mu_j$ and $\mu_m = \mu_i$ in (5.11)). We then obtain $|g - \mu_i| \geq \epsilon_{\text{tol}}^p / (2(1 + \Lambda_M)L)$ by (5.11).

We finally note that either, $a = \mu_i$ or a resides to the left of μ_i . We thus obtain

$$|g - a| \geq |g - \mu_i| \geq \frac{\epsilon_{\text{tol}}^p}{2(1 + \Lambda_M)L} \geq \frac{\epsilon_{\text{tol}}^p}{2(1 + \Lambda_M^{\max})L}. \quad (5.12)$$

The result for $|b - g|$ is identical and invokes analogous arguments.

The remainder of the argument is now identical to the argument for the proof of Proposition 1. \square

6 Computational cost

We now discuss the computational cost associated with the hp -EIM approaches presented above. We discuss the cost for the two methods concurrently since the separation of the computations in offline and online stages is very similar.

In the hp -EIM offline stage, we perform h - and p -refinement: parameter domain partition and construction of EIM spaces and EIM interpolation nodes restricted to each parameter subdomain. The offline stage is expensive, since the cost depends on the (typically large) number \mathcal{N} of vertices in the triangulation $\mathcal{T}^{\mathcal{N}}(\Omega)$. In particular, if we assume that the generated partition has K subdomains, we must perform KMN function evaluations in order to construct an EIM space of dimension M associated with each subdomain. For the GC approach, we must also perform MN function evaluations for each intermediate space associated with an intermediate subdomain; for this reason we expect the GC approach to be more expensive than the AP approach in the offline stage.

In the hp -EIM online stage, given any new parameter value $\mu \in \mathcal{D}$, we first determine to which subdomain $\mathcal{V}_{k^*} \subset \mathcal{D}$ the new parameter value belongs. For the AP approach this search can be performed at cost $\mathcal{O}(\log_2(K))$ for K subdomains since the subdomains can be organized in a binary tree: at each level in the tree a comparison between the distances from μ to two anchor points determines whether to proceed to the left or to the right branch. For the GC approach this search can be performed at cost $\mathcal{O}(\log_{2^P}(K))$ since the subdomains can be organized in a tree of order 2^P : at each level in the tree an elementwise comparison between μ and the gravity center g determines to which of the 2^P branches to proceed at the next level. We note that for the same number of subdomains and for $P > 1$ parameters, we expect that the GC approach yields the more efficient search since the tree has fewer levels and the cost at each level is roughly the same — $\mathcal{O}(P)$ for both approaches. However the cost of this search is in any event typically negligible.

Once the correct subdomain \mathcal{V}_{k^*} that contains the given parameter value μ has been determined, we perform the standard EIM online stage: we solve a system of the form (3.2) at cost $\mathcal{O}(M^2)$. The key point is that function evaluations of $\mathcal{G}(\cdot; \mu)$ are required only at the M spatial interpolation nodes in T^{k^*} . Of course, if we wish to additionally visualize $\mathcal{G}_M(x; \mu)$ for all $x \in \Omega$, the cost becomes \mathcal{N} -dependent.

We emphasize that our hp -EIM procedures provide a reduction in the $\mathcal{O}(M^2)$ *online* computational cost through reduction in the number of EIM basis functions, M . A smaller M requires a larger number of subdomains, K . However,

the reduction in M does not balance the increase in K : the product KM increases with K . As a result the hp -EIM *offline* stage is more expensive than than the standard EIM offline stage.

We finally note that an alternative “discretely orthogonal” basis for W_M is $\{V_m^M, 1 \leq m \leq M\}$. This basis enables $\mathcal{O}(M)$ -complexity computation of the EIM approximation since $q_j(t_i)$ in (3.2) is in this case replaced by δ_{ji} . However, this basis is not hierarchical since $\{V_m^{M-1}, 1 \leq m \leq M-1\} \not\subseteq \{V_m^M, 1 \leq m \leq M\}$ and hence the computation of the characteristic functions would have to be computed as an additional final step in the EIM precomputation procedure. In any event, when the EIM is applied within the reduced basis framework, the computational cost of the RB online stage scales as M^2 independent of the choice of the EIM basis [7]. For this reason, and for simplicity of exposition, we consider in this paper the standard EIM basis functions $q_m, 1 \leq m \leq M$.

7 Numerical results

We present in this section numerical results for our two hp -EIM approaches applied to two model problems. In all cases, the hp -EIM yields significant (online) speedup compared to the standard EIM.

7.1 Example 1: 2D Gaussian surface

We define the spatial domain $\Omega \equiv (0, 1) \times (0, 1) \subset \mathbb{R}^2$, and we introduce a triangulation $\mathcal{T}^{\mathcal{N}}(\Omega)$ with $\mathcal{N} = 2601$ vertices. We define the parameter domain $\mathcal{D} \equiv [0.3, 0.7] \times [0.3, 0.7] \subset \mathbb{R}^2$, and we introduce a “tensor-product” train sample $\Xi_{\text{train}} \subset \mathcal{D}$ of size 1600. We consider the Gaussian function

$$\mathcal{G}(x; \mu) \equiv \exp\left(-\frac{(x_{(1)} - \mu_{(1)})^2}{0.02} - \frac{(x_{(2)} - \mu_{(2)})^2}{0.02}\right), \quad (7.1)$$

for $x = (x_{(1)}, x_{(2)}) \in \Omega$ and $\mu = (\mu_{(1)}, \mu_{(2)}) \in \mathcal{D}$. The function \mathcal{G} is thus parametrized by the location of the Gaussian source.

Clearly, \mathcal{G} is particularly well suited for hp -adaptivity: snapshots associated with μ in one region of \mathcal{D} do not provide a good approximation for functions associated with μ in another region of \mathcal{D} . We thus expect an hp -EIM procedure to provide significant reduction in M for this example.

To obtain a benchmark for comparison, we first pursue the standard EIM with $M_{\text{max}} = 196$, which corresponds to a tolerance 10^{-8} satisfied over Ξ_{train} .

Parameter Multi-Domain “hp” Empirical Interpolation

	M_{\max}	ϵ_{tol}^h	$K(M_{\max}, \epsilon_{\text{tol}}^h)$
Computation 1	143	1	2
Computation 2	77	0.99	12
Computation 3	40	0.8	55
Computation 4	30	0.6	106

Table 1: Specified M_{\max} and ϵ_{tol}^h , and the required number of subdomains $K(M_{\max}, \epsilon_{\text{tol}}^h)$ for the anchor point procedure applied to Example 1.

	M_{\max}	$K(M_{\max})$
Computation 1	99	4
Computation 2	58	16
Computation 3	38	64
Computation 4	27	256

Table 2: Specified M_{\max} and the required number of subdomains $K(M_{\max})$ for the gravity center procedure applied to Example 1.

We note that the standard EIM is a special case both of the AP procedure and of the GC procedure for $K = 1$ subdomain.

We next pursue the AP splitting procedure. We specify $\epsilon_{\text{tol}}^p = 10^{-8}$ as the tolerance to be satisfied over the training set on each subdomain. We then perform four computations for different M_{\max} and ϵ_{tol}^h and obtain partitions with $K(M_{\max}, \epsilon_{\text{tol}}^h)$ subdomains as listed in Table 1. In Figure 3(a) we show the maximum error during each of the four computations,

$$\epsilon_{M_{\max}}^{K, \text{AP}} = \max_{\mu \in \tilde{\Xi}_{\text{train}}^{\text{AP}}} \|\mathcal{G}_{M, K}^{\text{AP}}(\cdot; \mu) - \mathcal{G}(\cdot; \mu)\|_{L^\infty}, \quad (7.2)$$

for $K = 2, 12, 55, 106$; here $\tilde{\Xi}_{\text{train}}^{\text{AP}}$ denotes the union of the train samples over each of the subdomains (we also show in Figure 3(a) the benchmark convergence for the case $K = 1$). In Figures 4(a) and 4(b) we show the partitions of \mathcal{D} with $K = 12$ and $K = 55$ subdomains, respectively. We note that the size of the subdomains is rather uniform over \mathcal{D} , which reflects the uniform parameter dependence of \mathcal{G} , as expected.

We then pursue the GC splitting procedure. We specify $\epsilon_{\text{tol}}^p = 10^{-8}$ as the tolerance to be satisfied over the training set on each subdomain. We then perform four computations for different M_{\max} and obtain partitions with $K(M_{\max})$ subdomains as listed in Table 2. In Figure 3(b) we show the maximum error

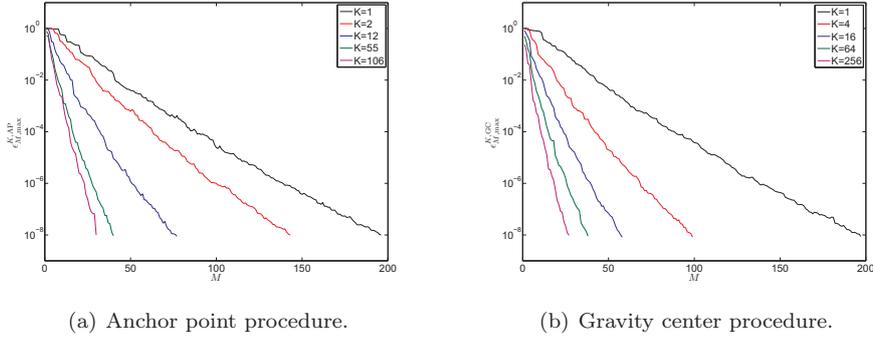


Figure 3: Convergence for Example 1.

during each of the four computations,

$$\epsilon_{M,\max}^{K,GC} = \max_{\mu \in \tilde{\Xi}_{\text{train}}^{\text{GC}}} \|\mathcal{G}_{M,K}^{\text{GC}}(\cdot; \mu) - \mathcal{G}(\cdot; \mu)\|_{L^\infty}, \quad (7.3)$$

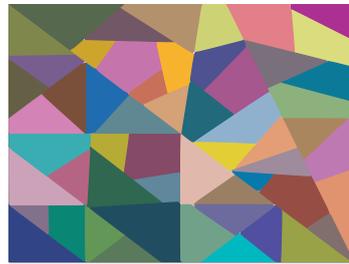
for $K = 4, 16, 64, 256$; here $\tilde{\Xi}_{\text{train}}^{\text{GC}}$ denotes the union of the train samples over each of the subdomains. In Figures 4(c) and 4(d) we show the partitions of \mathcal{D} with $K = 16$ and $K = 64$ subdomains, respectively. We note that the size of the subdomains is uniform.

We finally compare in Figure 5 our two approaches in terms of the number of required subdomains K for specified M_{\max} such that $\epsilon_{\text{tol}}^p = 10^{-8}$ is satisfied over train samples over all subdomains. We note that there is an algebraic relationship between K and M_{\max} , and that for Example 1 the two approaches perform very similarly in terms of the number of subdomains required for a specified tolerance. We further note that the product KM_{\max} increases with K , and thus a smaller M_{\max} yields larger offline cost. However, we would expect less steep curves in Figure 5 had we decreased the half-width of the Gaussian: a narrower Gaussian would have even more local parameter dependence and hence benefit more from hp -treatment.

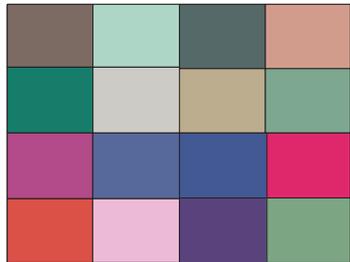
Parameter Multi-Domain “hp” Empirical Interpolation



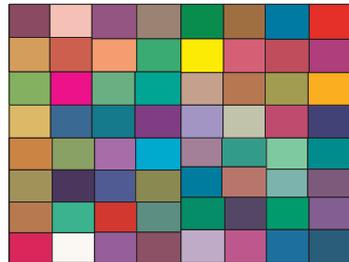
(a) Anchor point procedure, $K = 12$.



(b) Anchor point procedure, $K = 55$.



(c) Gravity center procedure, $K = 16$.



(d) Gravity center procedure, $K = 64$.

Figure 4: Parameter domain partitions for Example 1.

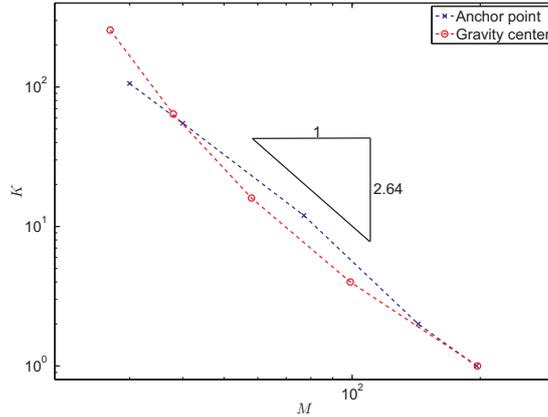


Figure 5: The number of required subdomains K as a function of specified M_{\max} for a given satisfied tolerance ϵ_{tol}^p .

7.2 Example 2: 3D wave function with near-singularity

Denote by $B_R(\mathcal{O})$ a ball in \mathbb{R}^3 with radius R and centered at the origin \mathcal{O} . Then, consider the scalar function

$$\mathcal{G}(x; \mu) = \frac{\cos(k|x - c(\mu)|)}{k|x - c(\mu)|}, \quad x = (x_{(1)}, x_{(2)}, x_{(3)}) \in \Omega = B_1(\mathcal{O}), \quad (7.4)$$

with $\mu = (k, r, \theta, \varphi) \in \mathcal{D}$ and $c(\mu) = r(\sin(\theta)\cos(\varphi), \sin(\theta)\sin(\varphi), \cos(\theta)) \in \mathbb{R}^3$. The parameter domain \mathcal{D} is defined by $\mathcal{D} = [1, 10] \times [1.1, 20] \times [0, \pi/2] \times [0, \pi/2]$. The spatial domain Ω and the parameter domain \mathcal{D} are discretized by $15 \times 15 \times 15$ and $8 \times 8 \times 8 \times 8$ tensorized grids, respectively, leading to discrete versions $\mathcal{T}^{\mathcal{N}}(\Omega)$ and Ξ_{train} , respectively.

We note that \mathcal{G} is particularly well suited for hp -adaptivity: the function has a very different structure for different wave numbers k and different locations of the pulse $c \in \mathbb{R}^3$. Snapshots with rapid and slow oscillations have little in common, and thus snapshots associated with k large contribute little to approximations of functions associated with k small, and vice versa. Similarly, snapshots with the singularity at c (outside but) close to Ω have high amplitude close to c and moderate amplitude elsewhere; such functions contribute little to the approximation of functions associated with c far from Ω , which have almost

Parameter Multi-Domain “hp” Empirical Interpolation

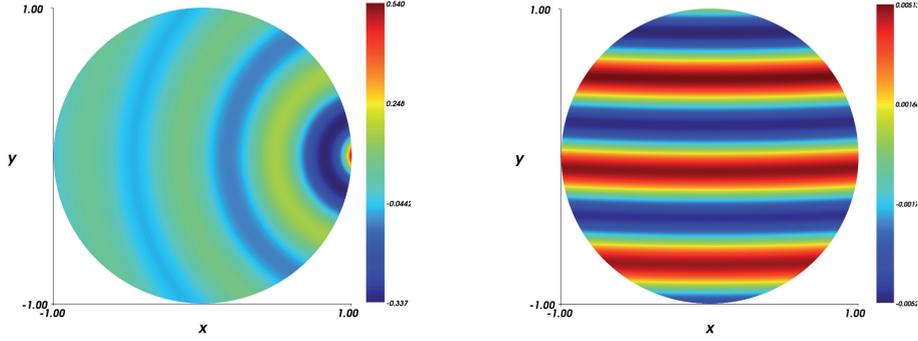


Figure 6: Example of the 3D wave function for a fixed $z = 0$ and parameter values $k = 10$, $c = (1.1, 0, 0)$ (left) and $k = 10$, $c = (0, 20, 0)$ (right).

	M_{\max}	ϵ_{tol}^h	$K(M_{\max}, \epsilon_{\text{tol}}^h)$
Computation 1	286	8	6
Computation 2	191	5	17
Computation 3	100	4	206

Table 3: Specified M_{\max} and ϵ_{tol}^h , and the required number of subdomains $K(M_{\max}, \epsilon_{\text{tol}}^h)$ for the anchor point procedure applied to Example 2.

constant amplitude. Two examples of \mathcal{G} for fixed $x_{(3)} = 0$ and parameter values $k = 10$, $c = (1.1, 0, 0)$ and $k = 10$, $c = (0, 20, 0)$ are shown in Figure 6; we note in particular the effect of c on the amplitude of the function.

To obtain a benchmark for comparison, we first pursue the standard EIM with $M_{\max} = 420$, which corresponds to a tolerance 10^{-3} satisfied over Ξ_{train} .

We next pursue the AP splitting procedure. We specify $\epsilon_{\text{tol}}^p = 10^{-3}$ as the tolerance to be satisfied over the training set on each subdomain. We then perform three computations for three different M_{\max} and ϵ_{tol}^h and obtain partitions with $K(M_{\max}, \epsilon_{\text{tol}}^h)$ subdomains as listed in Table 3. In Figure 7(a) we show the maximum error during each of the three computations, $\epsilon_{M, \max}^{K, \text{AP}}$, for $K = 6, 17, 206$.

We then pursue the GC splitting procedure. We specify $\epsilon_{\text{tol}}^p = 10^{-3}$ as the tolerance to be satisfied over the training set on each subdomain. We then perform three computations for four different M_{\max} and obtain partitions with

	M_{\max}	$K(M_{\max})$
Computation 1	301	16
Computation 2	238	76
Computation 3	200	151
Computation 4	146	676

Table 4: Specified M_{\max} and the required number of subdomains $K(M_{\max})$ for the gravity center procedure applied to Example 2.

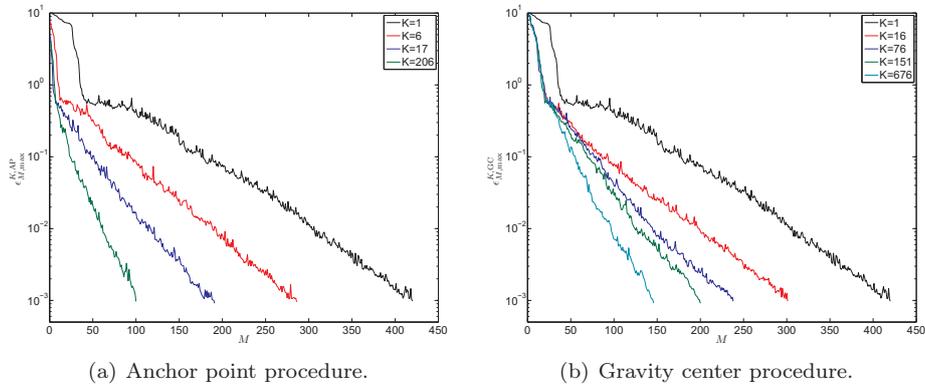


Figure 7: Convergence for Example 2.

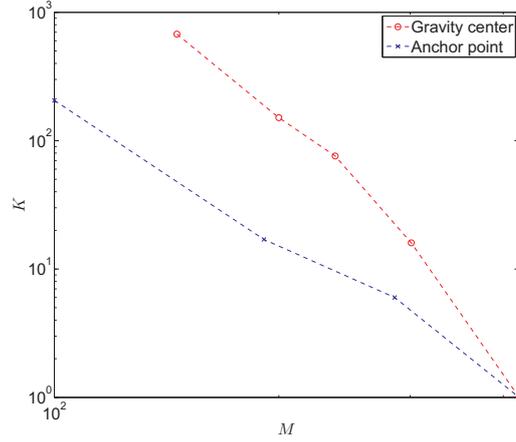


Figure 8: The number of required subdomains K as a function of specified M_{\max} for a given satisfied tolerance ϵ_{tol}^p .

$K(M_{\max})$ subdomains as listed in Table 4. In Figure 7(b) we show the maximum error during each of the four computations, $\epsilon_{M, \max}^{K, \text{GC}}$ for $K = 16, 76, 151, 676$.

We finally compare in Figure 8 our two approaches in terms of the number of required subdomains K for specified M_{\max} such that $\epsilon_{\text{tol}}^p = 10^{-3}$ is satisfied over the training sample on each subdomain. We note that there is an algebraic relationship between K and M , and that for Example 2 the AP approach seems to provide the somewhat more optimal partition.

8 Closing remarks

The hp -EIM procedures derived in this paper provide a partition of the full parameter domain into parameter subdomains; a standard EIM approximation is pursued on each subdomain in order to satisfy a specified tolerance ϵ_{tol}^p for a specified maximum number M_{\max} of EIM basis functions. Two different approaches are discussed. The first approach — the anchor point splitting procedure — is based on the first two modes associated with a standard EIM approximation: a given parameter (sub)domain is split into two new subdomains by a hyperplane equidistant from the the first two parameter values identified by the standard EIM Greedy sampling procedure. The second approach — the

gravity center splitting procedure — is based on all M_{\max} modes associated with a standard EIM approximation: a given parameter (sub)domain is split into 2^P tensorized new subdomains at the gravity center of the M_{\max} parameter values identified by the EIM Greedy sampling procedure (recall that P is the dimension of the parameter domain). For both approaches, *a priori* convergence theory guarantees successful termination of the partition process.

Through two numerical examples we demonstrate that both the AP and GC approaches provide significant computational speedup (approximation space dimension reduction) in the EIM online stage through reduction in the required EIM space dimension. Admittedly, our two examples are particularly well suited for *hp*-treatment. Functions with very smooth parameter dependence will be less well suited for *hp*-treatment. In this case we expect that the required number of subdomains for specified dimension reduction (and specified tolerance) is large, and hence the offline cost might be large.

The AP approach seems to be somewhat better suited for higher dimensional parameter domains in particular when the parameter dependence of the function under consideration is anisotropic: only two new subdomains are introduced for each splitting. The GC approach is on the other hand arguably simpler in terms of implementation; in particular, the tensorized subdomain structure enables explicit construction of the parameter training sets associated with each subdomain.

A straightforward application of the *hp*-EIM procedures is within the reduced basis (RB) framework for order reduction of non-affine parametrized partial differential equations. In this context, the (*hp*-EIM or) EIM accommodates efficient offline-online computational procedures through affine approximations of the non-affine differential operator [1, 6, 7, 8, 11]. The cost of the RB online stage grows quadratically with the number of terms in the affine approximation of the operator (M) and hence the *hp*-EIM approach will reduce the RB online cost. Similarly, the *hp*-EIM approach may be applied within the related *hp*-RB framework [4]. In [5] the gravity center approach discussed in this paper is applied within an RB framework for the electric field integral equation.

The *hp*-EIM method may also provide an improvement of the rigorous *a posteriori* error bounds recently introduced for the EIM [2]. Currently these bounds are global in parameter, and the *hp*-EIM thus provides a natural way of localizing, and hence in effect sharpening, the bounds.

Acknowledgements

We are grateful for many fruitful discussions with Prof. Anthony T. Patera and Prof. Martin A. Grepl. This work has been supported by the Norwegian University of Science and Technology, University of California, Berkeley, Lawrence Berkeley National Laboratory, and OSD/AFOSR Grant Number FA9550-09-1-0613.

Bibliography

- [1] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [2] J. L. Eftang, M. A. Grepl, and A. T. Patera. A posteriori error bounds for the empirical interpolation method. *Comptes Rendus Mathematique*, 348(9-10):575 – 579, 2010.
- [3] J. L. Eftang, D. J. Knezevic, and A. T. Patera. An *hp* certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems, to appear*, 2011. <http://augustine.mit.edu>.
- [4] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An “*hp*” certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200, 2010.
- [5] M. Fares, J. S. Hesthaven, Y. Maday, and B. Stamm. The reduced basis method for the electric field integral equation. *Journal of Computational Physics, in press*, 2011.
- [6] M. Grepl. *A Posteriori* Error Bounds for Reduced-Basis Approximations of Nonaffine and Nonlinear Parabolic Partial Differential Equations. *Mathematical Models and Methods in Applied Sciences, submitted*, 2010.
- [7] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.

Paper 5

- [8] A. E. Løvgren, Y. Maday, and E. M. Rønquist. The reduced basis element method: Offline-online decomposition in the nonconforming, nonaffine case. In J. S. Hesthaven and E. M. Rønquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, volume 76 of *Lecture Notes in Computational Science and Engineering*, pages 247–254. Springer Berlin Heidelberg, 2011.
- [9] Y. Maday, N. C. Nguyen, A. T. Patera, and G. S. H. Pau. A general multipurpose interpolation procedure: The magic points. *Communications in Pure and Applied Mathematics*, 8:383–404, 2009.
- [10] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*, volume 37 of *Texts Appl. Math.* Springer, New York, 1991.
- [11] G. Rozza. Reduced basis methods for stokes equations in domains with non-affine parameter dependence. *Computing and Visualization in Science*, 12:23–35, 2009.
- [12] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

PAPER 6

***A POSTERIORI* ERROR BOUNDS FOR THE
EMPIRICAL INTERPOLATION METHOD**

JENS L. EFTANG, MARTIN A. GREPL, AND ANTHONY T. PATERA

Published in
Comptes Rendus Mathematique,
Vol. 348, No. 9-10 (2010), pp. 575-579

A *POSTERIORI* ERROR BOUNDS FOR THE EMPIRICAL INTERPOLATION METHOD

JENS L. EFTANG*, MARTIN A. GREPL†, AND ANTHONY T. PATERA‡

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

*†Numerical Mathematics,
RWTH Aachen University,
Aachen, Germany*

*‡Department of Mechanical Engineering,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA*

Abstract

We present rigorous *a posteriori* error bounds for the Empirical Interpolation Method (EIM). The essential ingredients are (i) analytical upper bounds for the parametric derivatives of the function to be approximated, (ii) the EIM “Lebesgue constant,” and (iii) information concerning the EIM approximation error at a finite set of points in parameter space. The bound is computed “offline” and is valid over the entire parameter domain; it is thus readily employed in (say) the “online” reduced basis context. We present numerical results that confirm the validity of our approach.

1 Introduction

The Empirical Interpolation Method (EIM), introduced in [1], serves to construct “affine” approximations of “non-affine” parametrized functions. The EIM is frequently applied in reduced basis approximation of parametrized partial differential equations with non-affine parameter dependence [6]; the affine approximation of the coefficient functions is crucial for computational efficiency. In previous work [1, 6] an estimator for the interpolation error is developed; this estimator is often very accurate, however it is not a rigorous upper bound. In this paper, we develop a rigorous *a posteriori* upper bound for the interpolation error and we present numerical results that confirm the validity of our approach.

To begin, we summarize the EIM [1, 6]. We are given a function $\mathcal{G} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ such that, for all $\mu \in \mathcal{D}$, $\mathcal{G}(\cdot; \mu) \in L^\infty(\Omega)$; here, $\mathcal{D} \subset \mathbb{R}^P$ is the parameter domain, $\Omega \subset \mathbb{R}^2$ is the spatial domain—a point in which shall be denoted by $x = (x_1, x_2)$ —and $L^\infty(\Omega) \equiv \{v \mid \text{ess sup}_{x \in \Omega} |v(x)| < \infty\}$. We introduce a finite train sample $\Xi_{\text{train}} \subset \mathcal{D}$ which shall serve as our \mathcal{D} surrogate, and a triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$ of Ω with \mathcal{N} vertices over which we shall in practice realize $\mathcal{G}(\cdot; \mu)$ as a piecewise linear function.

We first define the nested EIM approximation spaces $W_M^{\mathcal{G}}$, $1 \leq M \leq M_{\text{max}}$. We first choose $\mu_1 \in \mathcal{D}$, compute $g_1 \equiv \mathcal{G}(\cdot; \mu_1)$, and define $W_1^{\mathcal{G}} \equiv \text{span}\{g_1\}$; then, for $2 \leq M \leq M_{\text{max}}$, we determine

$$\mu_M \equiv \arg \max_{\mu \in \Xi_{\text{train}}} \inf_{z \in W_{M-1}^{\mathcal{G}}} \|\mathcal{G}(\cdot; \mu) - z\|_{L^\infty(\Omega)}, \quad (1.1)$$

compute $g_M \equiv \mathcal{G}(\cdot; \mu_M)$, and define $W_M^{\mathcal{G}} \equiv \text{span}\{g_m\}_{m=1}^M$.

We next define the nested set of EIM interpolation nodes $T_M^{\mathcal{G}} \equiv \{t_1, \dots, t_M\}$, $1 \leq M \leq M_{\text{max}}$. We first set $t_1 \equiv \arg \sup_{x \in \Omega} |g_1(x)|$ and $q_1 \equiv g_1/g_1(t_1)$; then, for $2 \leq M \leq M_{\text{max}}$, we solve the linear system

$$\sum_{j=1}^{M-1} \omega_j^{M-1} q_j(t_i) = g_M(t_i), \quad 1 \leq i \leq M-1, \quad (1.2)$$

and set $r_M(x) = g_M(x) - \sum_{j=1}^{M-1} \omega_j^{M-1} q_j(x)$,

$$t_M \equiv \arg \sup_{x \in \Omega} |r_M(x)|, \quad (1.3)$$

and $q_M = r_M/r_M(t_M)$. For $1 \leq M \leq M_{\text{max}}$, we define the matrix $B^M \in \mathbb{R}^{M \times M}$ such that $B_{ij}^M \equiv q_j(t_i)$, $1 \leq i, j \leq M$; we note that B^M is lower triangular with unity diagonal and that $\{q_m\}_{m=1}^M$ is a basis for $W_M^{\mathcal{G}}$ [1, 6].

We are now given a function $\mathcal{H} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ such that, for all $\mu \in \mathcal{D}$, $\mathcal{H}(\cdot; \mu) \in L^\infty(\Omega)$. We define for any $\mu \in \mathcal{D}$ the EIM interpolant $\mathcal{H}_{W_M^{\mathcal{G}}}(\cdot; \mu) \in W_M^{\mathcal{G}}$ as the interpolant of $\mathcal{H}(\cdot; \mu)$ over the set $T_M^{\mathcal{G}}$. Specifically

$$\mathcal{H}_{W_M^{\mathcal{G}}}(\cdot; \mu) \equiv \sum_{m=1}^M \phi_{M m}(\mu) q_m, \quad (1.4)$$

where

$$\sum_{j=1}^M B_{ij}^M \phi_{M j}(\mu) = \mathcal{H}(t_i; \mu), \quad 1 \leq i \leq M. \quad (1.5)$$

Note that the determination of the coefficients $\phi_{Mm}(\mu)$ requires only $\mathcal{O}(M^2)$ computational cost.

Finally, we define a ‘‘Lebesgue constant’’ [7] $\Lambda_M \equiv \sup_{x \in \Omega} \sum_{m=1}^M |V_m^M(x)|$, where $V_m^M \in W_M^{\mathcal{G}}$ are the characteristic functions of $W_M^{\mathcal{G}}$ satisfying $V_m^M(t_n) \equiv \delta_{mn}$, $1 \leq m, n \leq M$; here, δ_{mn} is the Kronecker delta symbol. We recall that (i) the set of all characteristic functions $\{V_m^M\}_{m=1}^M$ is a basis for $W_M^{\mathcal{G}}$, and (ii) the Lebesgue constant Λ_M satisfies $\Lambda_M \leq 2^M - 1$ [1, 6]. In applications, the actual asymptotic behavior of Λ_M is much better, as we shall observe subsequently.

2 A Posteriori Error Estimation

We now develop the new and rigorous upper bound for the error associated with the empirical interpolation of a function $\mathcal{F} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$. We shall assume that \mathcal{F} is parametrically smooth; for simplicity here, we suppose $\mathcal{F} \in C^\infty(\mathcal{D}, L^\infty(\Omega))$. Our bound depends on the parametric derivatives of \mathcal{F} and on the EIM interpolant of these derivatives. For this reason, we introduce a multi-index of dimension P , $\beta \equiv (\beta_1, \dots, \beta_P)$, where the β_i , $1 \leq i \leq P$, are non-negative integers; we further define the length $|\beta| \equiv \sum_{i=1}^P \beta_i$, and denote the set of all distinct multi-indices β of dimension P of length I by \mathcal{M}_I^P . The cardinality of \mathcal{M}_I^P is given by $\text{card}(\mathcal{M}_I^P) = \binom{P+I-1}{I}$. For any multi-index β , we define

$$\mathcal{F}^{(\beta)}(x; \mu) \equiv \frac{\partial^{|\beta|} \mathcal{F}}{\partial \mu_{(1)}^{\beta_1} \dots \partial \mu_{(P)}^{\beta_P}}(x; \mu); \quad (2.1)$$

we require that $\max_{\mu \in \mathcal{D}} \max_{\beta \in \mathcal{M}_p^P} \|\mathcal{F}^{(\beta)}(\cdot; \mu)\|_{L^\infty(\Omega)} \leq \sigma_p (< \infty)$ for integer $p \geq 0$.

Given any $\mu \in \mathcal{D}$, we define for $1 \leq M \leq M_{\max}$ the interpolants of $\mathcal{F}(\cdot; \mu)$ and $\mathcal{F}^{(\beta)}(\cdot; \mu)$ as $\mathcal{F}_M(\cdot; \mu) \equiv \mathcal{F}_{W_M^{\mathcal{F}}}(\cdot; \mu)$ and $(\mathcal{F}^{(\beta)})_M(\cdot; \mu) \equiv \mathcal{F}_{W_M^{\mathcal{F}^{(\beta)}}}(\cdot; \mu)$, respectively. We emphasize that both interpolants $\mathcal{F}_M(\cdot; \mu)$ and $(\mathcal{F}^{(\beta)})_M(\cdot; \mu)$ lie in the same space $W_M^{\mathcal{F}}$ —we do not introduce a separate space, $W_M^{\mathcal{F}^{(\beta)}}$, spanned by solutions of $\mathcal{F}^{(\beta)}(\cdot; \mu_M)$, $1 \leq M \leq M_{\max}$. It is thus readily demonstrated that $(\mathcal{F}^{(\beta)})_M(\cdot; \mu) = (\mathcal{F}_M)^{(\beta)}(\cdot; \mu)$, which we thus henceforth denote $\mathcal{F}_M^{(\beta)}(\cdot; \mu)$.¹ Note that $\mathcal{F}_M^{(\beta)}(\cdot; \mu) \in W_M^{\mathcal{F}}$ is the unique interpolant satisfying

¹Let $Z_q = [q_1 \dots q_M]$ and $\bar{t}_M = [t_1 \dots t_M]$. We then have $\mathcal{F}_M(\cdot; \mu) = Z_q(B^M)^{-1} \mathcal{F}(\bar{t}_M; \mu)$ and $(\mathcal{F}^{(\beta)})_M(\cdot; \mu) = Z_q(B^M)^{-1} \mathcal{F}^{(\beta)}(\bar{t}_M; \mu)$. Since B^M and the basis functions q_i , $1 \leq i \leq M$, are independent of μ , it follows that $(\mathcal{F}_M)^{(\beta)}(\cdot; \mu) = (Z_q(B^M)^{-1} \mathcal{F}(\bar{t}_M; \mu))^{(\beta)} = Z_q(B^M)^{-1} \mathcal{F}^{(\beta)}(\bar{t}_M; \mu) = (\mathcal{F}^{(\beta)})_M(\cdot; \mu)$.

$\mathcal{F}_M^{(\beta)}(t_m; \mu) = \mathcal{F}^{(\beta)}(t_m; \mu)$, $1 \leq m \leq M$. We can further demonstrate [3] in certain cases that if $\mathcal{F}_M(\cdot; \mu)$ tends to $\mathcal{F}(\cdot; \mu)$ as $M \rightarrow \infty$ then $\mathcal{F}_M^{(\beta)}(\cdot; \mu)$ tends to $\mathcal{F}^{(\beta)}(\cdot; \mu)$ as $M \rightarrow \infty$.

We now develop the interpolation error upper bound. To begin, we introduce a set of points $\Phi \subset \mathcal{D}$ of size n_Φ and define $\rho_\Phi \equiv \max_{\mu \in \mathcal{D}} \min_{\tau \in \Phi} |\mu - \tau|$; here $|\cdot|$ is the usual Euclidean norm. We then define

$$\delta_{M,p} \equiv (1 + \Lambda_M) \frac{\sigma_p}{p!} \rho_\Phi^p P^{p/2} + \sup_{\tau \in \Phi} \left(\sum_{j=0}^{p-1} \frac{\rho_\Phi^j}{j!} P^{j/2} \max_{\beta \in \mathcal{M}_j^P} \|\mathcal{F}^{(\beta)}(\cdot; \tau) - \mathcal{F}_M^{(\beta)}(\cdot; \tau)\|_{L^\infty(\Omega)} \right). \quad (2.2)$$

We can now demonstrate

Proposition 1. *For given positive integer p ,*

$$\max_{\mu \in \mathcal{D}} \|\mathcal{F}(\cdot; \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} \leq \delta_{M,p}, \quad (2.3)$$

$\forall \mu \in \mathcal{D}$, $1 \leq M \leq M_{\max}$.

Proof. We present the proof for $P = 1$ and refer the reader to [2] for the general case $P \geq 1$. For brevity, we first define (assuming existence) $\mathcal{A}_\mathcal{G}^p(\tau, \mu) \equiv \sum_{j=0}^{p-1} \mathcal{G}^{(j)}(\cdot; \tau) \frac{(\mu - \tau)^j}{j!}$ as the first p terms in the Taylor series of \mathcal{G} around τ . We then choose τ as $\tau^*(\mu) \equiv \arg \min_{\tau \in \Phi} |\mu - \tau|$. We note that

$$\begin{aligned} \|\mathcal{F}(\cdot; \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} &\leq \|\mathcal{F}(\cdot; \mu) - \mathcal{A}_\mathcal{F}^p(\tau^*, \mu)\|_{L^\infty(\Omega)} \\ &\quad + \|\mathcal{A}_\mathcal{F}^p(\tau^*, \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} \end{aligned} \quad (2.4)$$

for all $\mu \in \mathcal{D}$. We recall the univariate Taylor series expansion with remainder in integral form $\mathcal{F}(x; \mu) = \mathcal{A}_\mathcal{F}^p(\tau, \mu) + \int_\tau^\mu \mathcal{F}^{(p)}(x; \bar{\tau}) \frac{(\mu - \bar{\tau})^{p-1}}{(p-1)!} d\bar{\tau}$. We can now bound the first term on the right hand side of (2.4) by

$$\|\mathcal{F}(\cdot; \mu) - \mathcal{A}_\mathcal{F}^p(\tau^*, \mu)\|_{L^\infty(\Omega)} \leq \left| \int_{\tau^*}^\mu \left\| \mathcal{F}^{(p)}(\cdot; \bar{\tau}) \frac{(\mu - \bar{\tau})^{p-1}}{(p-1)!} \right\|_{L^\infty(\Omega)} d\bar{\tau} \right| \leq \frac{\sigma_p}{p!} \rho_\Phi^p \quad (2.5)$$

A Posteriori Error Bounds for the EIM

for all $\mu \in \mathcal{D}$. For the second term in (2.4), we obtain

$$\begin{aligned} \|\mathcal{A}_{\mathcal{F}}^p(\tau^*, \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} &\leq \|\mathcal{A}_{\mathcal{F}}^p(\tau^*, \mu) - \mathcal{A}_{\mathcal{F}_M}^p(\tau^*, \mu)\|_{L^\infty(\Omega)} \\ &\quad + \|\mathcal{A}_{\mathcal{F}_M}^p(\tau^*, \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} \end{aligned} \quad (2.6)$$

for all $\mu \in \mathcal{D}$. For the first term in (2.6) we note that

$$\begin{aligned} &\|\mathcal{A}_{\mathcal{F}}^p(\tau^*, \mu) - \mathcal{A}_{\mathcal{F}_M}^p(\tau^*, \mu)\|_{L^\infty(\Omega)} \\ &\leq \sup_{\tau \in \Phi} \left(\sum_{j=0}^{p-1} \frac{\rho_\Phi^j}{j!} \|\mathcal{F}^{(j)}(\cdot; \tau) - \mathcal{F}_M^{(j)}(\cdot; \tau)\|_{L^\infty(\Omega)} \right), \quad \forall \mu \in \mathcal{D}. \end{aligned} \quad (2.7)$$

From the definition of the characteristic functions V_m^M , we obtain

$$\begin{aligned} &\sum_{j=0}^{p-1} \mathcal{F}_M^{(j)}(x; \tau^*) \frac{(\mu - \tau^*)^j}{j!} - \mathcal{F}_M(x; \mu) \\ &= \sum_{m=1}^M \left[\sum_{j=0}^{p-1} \mathcal{F}_M^{(j)}(t_m; \tau^*) \frac{(\mu - \tau^*)^j}{j!} - \mathcal{F}_M(t_m; \mu) \right] V_m^M(x). \end{aligned} \quad (2.8)$$

We then invoke the interpolation property (for any non-negative integer j) $\mathcal{F}_M^{(j)}(t_m; \mu) = \mathcal{F}^{(j)}(t_m; \mu)$, $1 \leq m \leq M$, and the definition of the Lebesgue constant Λ_M , to bound the second term in (2.6) by

$$\begin{aligned} &\|\mathcal{A}_{\mathcal{F}_M}^p(\tau^*, \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)} \\ &\leq \|\mathcal{A}_{\mathcal{F}}^p(\tau^*, \mu) - \mathcal{F}(\cdot; \mu)\|_{L^\infty(\Omega)} \Lambda_M \leq \frac{\sigma_p}{p!} \rho_\Phi^p \Lambda_M, \quad \forall \mu \in \mathcal{D}. \end{aligned} \quad (2.9)$$

The desired result (for $P = 1$) directly follows. \square

We make several remarks concerning this result. First, we may choose p such that the two terms in (2.2) balance—a higher p will reduce the contribution of the first term but will increase the contribution of the second term. Second, we note that the bound $\delta_{M,p}$ is μ -independent. We can readily develop a μ -dependent bound by replacing ρ_Φ with the actual distance between μ and the closest $\tau \in \Phi$; this μ -dependent bound can serve (i) to adaptively construct an economical point set Φ , and (ii) to replace the true (expensive) error in the

greedy identification of the EIM spaces $W_M^{\mathcal{G}}$. Third, we can increase the sharpness of the bound by localizing the derivative bounds σ_p : this is best achieved through an “ hp ” approach for the EIM; we note that the “ hp ” framework developed in [4] for the reduced basis method readily adapts to the EIM (see also [5] for an alternative approach). Fourth, we note that in the “limit” $\rho_{\Phi} \rightarrow 0$ the effectivity of the bound approaches unity; of course, we will never in practice let $\rho_{\Phi} \rightarrow 0$ because this implies the computation of the interpolant at every point in \mathcal{D} . Fifth, we note that our bound at no point requires computation of spatial derivatives of the function to be approximated.

We conclude this section by summarizing the computational cost associated with $\delta_{M,p}$. We assume that the bounds σ_p can be obtained analytically. We compute Λ_M in $\mathcal{O}(M^2\mathcal{N})$ operations, and we compute the interpolation errors $\|\mathcal{F}^{(\beta)}(\cdot; \tau) - \mathcal{F}_M^{(\beta)}(\cdot; \tau)\|_{L^\infty(\Omega)}$, $0 < |\beta| < p - 1$, for all $\tau \in \Phi$, in $\mathcal{O}(n_{\Phi}M\mathcal{N}) \sum_{j=0}^{p-1} \text{card}(\mathcal{M}_j^P)$ operations (we assume $M \ll \mathcal{N}$); certainly the growth of \mathcal{M}_p^P will preclude large P . Note the computational cost is “offline” only—the bound is valid for all $\mu \in \mathcal{D}$.

3 Numerical Results

We shall consider the empirical interpolation of a Gaussian function $\mathcal{F}(\cdot; \mu)$ over two different parameter domains $\mathcal{D} = \mathcal{D}_I$ and $\mathcal{D} = \mathcal{D}_{II}$. The spatial domain is $\Omega \equiv [0, 1]^2$; we introduce a triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$ with $\mathcal{N} = 2601$ vertices. We shall compare our bound with the true interpolation error over the parameter domain. To this end, we define the maximum error $\varepsilon_M \equiv \max_{\mu \in \Xi_{\text{train}}} e_M(\mu)$ and the average effectivity $\bar{\eta}_{M,p} \equiv \text{mean}_{\mu \in \Xi_{\text{test}}} \delta_{M,p}/e_M(\mu)$; here, $e_M(\mu) \equiv \|\mathcal{F}(\cdot; \mu) - \mathcal{F}_M(\cdot; \mu)\|_{L^\infty(\Omega)}$, and $\Xi_{\text{test}} \subset \mathcal{D}$ is a test sample of finite size $n_{\Xi_{\text{test}}}$.

We first consider the case $\mathcal{D} = \mathcal{D}_I \equiv [0.1, 1]$ and hence $P = 1$; we let

$$\mathcal{F}(x; \mu) = \mathcal{F}_I(x; \mu) \equiv \exp\left(\frac{-(x_1 - 0.5)^2 - (x_2 - 0.5)^2}{2\mu^2}\right). \quad (3.1)$$

We introduce an equidistant train sample $\Xi_{\text{train}} \subset \mathcal{D}$ of size 500; we take $\mu_1 = 1$ and pursue the EIM with $M_{\text{max}} = 12$. In Figure 1 we report ε_M and $\delta_{M,p}$, $p = 1, 2, 3, 4$, for $1 \leq M \leq M_{\text{max}}$; we consider $n_{\Phi} = 41$ and $n_{\Phi} = 141$ ($\rho_{\Phi} = 1.125\text{E-}2$ and $\rho_{\Phi} \approx 3.21\text{E-}3$, respectively). We observe that the error bounds initially decrease, but then “plateau” in M . The bounds are very sharp for sufficiently small M , but eventually the first term in (2.2) dominates and compromises the sharpness of the bounds; for larger p , the bound is better

A Posteriori Error Bounds for the EIM

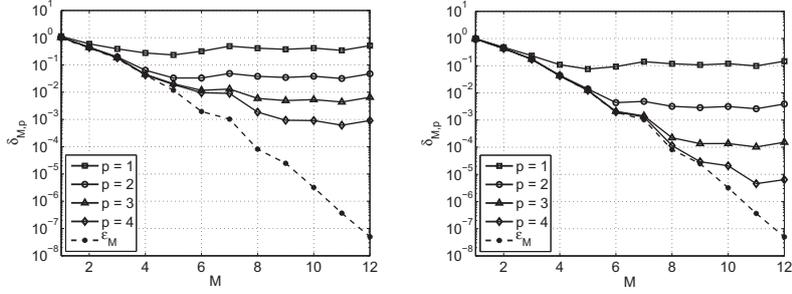


Figure 1: Error bounds $\delta_{M,p}$ for $P = 1$ and $p = 1, 2, 3, 4$ with $n_\Phi = 41$ (left) and $n_\Phi = 141$ (right).

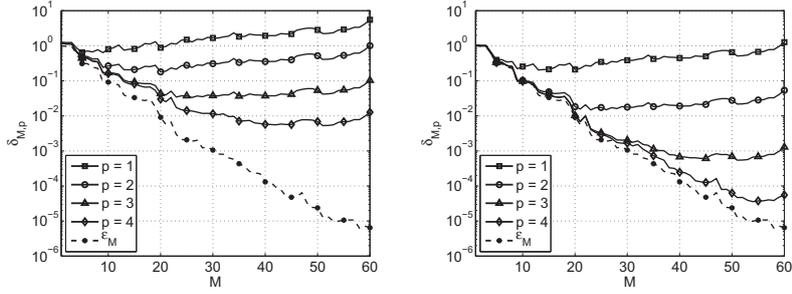


Figure 2: Error bounds $\delta_{M,p}$ for $P = 2$ and $p = 1, 2, 3, 4$ with $n_\Phi = 100$ (left) and $n_\Phi = 1600$ (right).

for a larger range of M . We find that $1 \leq \Lambda_M \leq 5.18$ for $1 \leq M \leq M_{\max}$ and, for the case $p = 4$ with $n_\Phi = 141$, $\bar{\eta}_{M,p} \sim \mathcal{O}(10)$ ($n_{\Xi_{\text{test}}} = 150$) except for large M . The modest growth of the Lebesgue constant is crucial to the good effectivity.

We next consider the case $\mathcal{D} = \mathcal{D}_{\text{II}} \equiv [0.4, 0.6]^2$ and hence $P = 2$; we introduce

$$\mathcal{F} = \mathcal{F}_{\text{II}}(x; \mu) = \exp\left(\frac{-(x_1 - \mu_{(1)})^2 - (x_2 - \mu_{(2)})^2}{2(0.1)^2}\right), \quad (3.2)$$

where $\mu \equiv (\mu_{(1)}, \mu_{(2)})$. We introduce a deterministic grid $\Xi_{\text{train}} \subset \mathcal{D}$ of size 1600; we take $\mu_1 = (0.4, 0.4)$ and pursue the EIM with $M_{\max} = 60$. In Figure 2 we

report ε_M and $\delta_{M,p}$, $p = 1, 2, 3, 4$, for $1 \leq M \leq M_{\max}$; we consider $n_\Phi = 100$ and $n_\Phi = 1600$ ($\rho_\Phi \approx 1.57\text{E-}2$ and $3.63\text{E-}3$, respectively). We observe the same behavior as for the $P = 1$ case: the errors initially decrease, but then “plateau” in M depending on the particular value of p . We find that $1 \leq \Lambda_M \leq 39.9$ and, for the case $p = 4$ with $n_\Phi = 1600$, $\bar{\eta}_{M,p} \sim \mathcal{O}(10)$ ($n_{\Xi_{\text{test}}} = 225$) for $1 \leq M \leq M_{\max}$.

Our results demonstrate that we can gainfully increase p —the number of terms in the Taylor series expansion—in order to reduce the role of the first term of $\delta_{M,p}$ and to limit the size of Φ . We also note that for the examples presented here the terms in the sum of (2.2) are well behaved, even though (for our $P = 2$ example in particular) it is not obvious that the space $W_M^{\mathcal{F}}$ contains good interpolants of the functions $\mathcal{F}^{(\beta)}(\cdot, \mu)$, $|\beta| \neq 0$.

Acknowledgements

We acknowledge Y. Maday and S. Boyaval for fruitful discussions and S. Sen and N. C. Nguyen for earlier contributions. The work was supported by AFOSR Grant No. FA9550-07-1-0425, OSD/AFOSR Grant No. FA9550-09-1-0613, the Norwegian University of Science and Technology, and the Excellence Initiative of the German federal and state governments.

Bibliography

- [1] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [2] J. L. Eftang, M. A. Grepl, and A. T. Patera. A posteriori error bounds for the empirical interpolation method. In preparation, 2011.
- [3] J. L. Eftang, M. A. Grepl, E. M. Rønquist, and A. T. Patera. Approximation of parametric derivatives by the empirical interpolation method. To be submitted to Foundations of Computational Mathematics. Norwegian University of Science and Technology, Numerics preprint 4/2011.
- [4] J. L. Eftang, A. T. Patera, and E. M. Rønquist. An “ hp ” certified reduced basis method for parametrized elliptic partial differential equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200, 2010.

A Posteriori Error Bounds for the EIM

- [5] M. Fares, J. S. Hesthaven, Y. Maday, and B. Stamm. The reduced basis method for the electric field integral equation. *Journal of Computational Physics*, *in press*, 2011.
- [6] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [7] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*, volume 37 of *Texts Appl. Math.* Springer, New York, 1991.

PAPER 7

**APPROXIMATION OF PARAMETRIC DERIVATIVES
BY THE EMPIRICAL INTERPOLATION METHOD**

JENS L. EFTANG, MARTIN A. GREPL, EINAR M. RØNQUIST, AND
ANTHONY T. PATERA

NTNU Preprint Numerics No. 4/2011

*To be submitted to
Foundations of Computational Mathematics,
2011*

APPROXIMATION OF PARAMETRIC DERIVATIVES BY THE EMPIRICAL INTERPOLATION METHOD

JENS L. EFTANG*, MARTIN A. GREPL†, EINAR M. RØNQUIST*,
AND ANTHONY T. PATERA‡

**Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway*

*†Numerical Mathematics,
RWTH Aachen University,
Aachen, Germany*

*‡Department of Mechanical Engineering,
Massachusetts Institute of Technology,
Cambridge, Massachusetts, USA*

Abstract

We introduce a general *a priori* convergence result for the approximation of parametric derivatives of parametrized functions. We show, with rather general assumptions on the particular approximation scheme, that the approximations of parametric derivatives of a given parametrized function are convergent provided that the approximation to the function itself is convergent. We present numerical results with one particular method for the approximation of parametrized functions — the Empirical Interpolation Method — to illustrate the general theory.

1 Introduction

We consider in this paper the approximation of *parametrized* functions, i.e., functions that in addition to spatial variables depend on one or several scalar parameters. In particular, we are concerned with the approximation of *parametric derivatives* of such functions, i.e., derivatives of parametrized functions with respect to the parameters. We develop a new convergence theory that demonstrates — with rather general assumptions on the particular approximation scheme — that the approximations of parametric derivatives of a given parametrized function are convergent provided that the approximation to the function itself is convergent.

The Empirical Interpolation Method (EIM), introduced in [1, 5], is an interpolation method specifically constructed for the approximation of parametrized functions.¹ The main focus of this paper is the EIM approximation of parametric derivatives of parametrized functions. The new convergence theory is developed with the EIM in mind, and is discussed and applied within the context of the EIM.

However, our new theoretical results here also apply to rather general approximation schemes *other* than the EIM; in particular, we may consider both projection-based and interpolation-based approximation. The main limitation of the theory is related to regularity assumptions in space and parameter on the parametrized function.

The results in this paper have several useful implications. First, if the EIM is employed for evaluation of an objective function subject to optimization with respect to a set of parameters, our theory shows that we may accurately compute the parametric Jacobian without expensive generation of additional EIM spaces, or alternatively finite difference Jacobian approximations. Second, the rigorous *a posteriori* bounds for the error in the EIM approximation recently introduced in [3] require computation of the EIM approximation of parametric derivatives at a finite number of points in the parameter domain; smaller EIM errors associated with these derivatives imply sharper EIM error bounds. This second point in particular motivates our work here.

The remainder of the paper is organized as follows. First, in Section 2 we introduce notation and recall some results from polynomial approximation theory. Next, in Section 3, we present the new general *a priori* convergence result. Then, in Section 4 we review the EIM and apply the new convergence theory in this particular context. Subsequently, in Section 5, we demonstrate the theory within the context of the EIM through numerical results. Finally, in Section 6 we provide some concluding remarks.

¹In particular, the EIM serves to construct parametrically affine approximations of parameter dependent non-affine or non-linear differential operators within the Reduced Basis (RB) framework for parametric reduced order modelling of partial differential equations [10]. An affine representation (or approximation) of the operator allows an efficient “offline-online” computational decoupling, which in turn is a crucial ingredient in the RB computational framework. We refer to [4, 5] for the application of the EIM for RB approximations.

2 Preliminaries

2.1 Notation

We denote by $\Omega \subset \mathbb{R}^d$ the spatial domain ($d = 1, 2, 3$); a particular point $x \in \Omega$ shall be denoted by $x = (x_{(1)}, \dots, x_{(d)})$. We denote by $\mathcal{D} \subset \mathbb{R}^P$ the parameter domain ($P \geq 1$); a particular parameter value $\mu \in \mathcal{D}$ shall be denoted by $\mu = (\mu_{(1)}, \dots, \mu_{(P)})$.

We introduce a sufficiently smooth function $\mathcal{F} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$. We suppose that $\mathcal{F}(\cdot; \mu) \in L^\infty(\Omega)$ for all $\mu \in \mathcal{D}$, and, for purposes of our theoretical arguments later, that $\mathcal{F}(x; \cdot) \in C^2(\mathcal{D})$ for all $x \in \Omega$. Here, $L^\infty(\Omega) = \{v : \text{ess sup}_{x \in \Omega} |v(x)| < \infty\}$ and $C^s(O)$ denotes the space of functions with continuous s -order derivatives over a domain O . We then introduce a multi-index of dimension P ,

$$\beta = (\beta_1, \dots, \beta_P), \quad (2.1)$$

where the entries β_i , $1 \leq i \leq P$, are non-negative integers. We define for any multi-index β the parametric derivatives of \mathcal{F} ,

$$\mathcal{F}^{(\beta)} = \frac{\partial^{|\beta|} \mathcal{F}}{\partial \mu_{(1)}^{\beta_1} \cdots \partial \mu_{(P)}^{\beta_P}}, \quad (2.2)$$

where

$$|\beta| = \sum_{i=1}^P \beta_i \quad (2.3)$$

is the length of β and hence the differential order. We denote the set of all distinct multi-indices β of dimension P of length p by \mathcal{M}_p^P .

For our theoretical arguments in Section 3 we shall write \mathcal{D} as the tensor product $\mathcal{D} = \mathcal{D}_{(1)} \times \cdots \times \mathcal{D}_{(P)}$, where $\mathcal{D}_{(i)} \subset \mathbb{R}$, $1 \leq i \leq P$. We shall further consider any particular parameter dimension $\mathcal{S} \equiv \mathcal{D}_j$, $1 \leq j \leq P$, and assume without loss of generality² that $\mathcal{S} = [-1, 1]$. In this case we fix the $P - 1$ parameter values $\mu_{(i)} \in \mathcal{D}_{(i)}$, $1 \leq i \leq P$, $i \neq j$, and we introduce the function $\mathcal{J}_{\beta,j} : \Omega \times \mathcal{S} \rightarrow \mathbb{R}$ defined for $x \in \Omega$ and $\kappa \in \mathcal{S}$ by

$$\mathcal{J}_{\beta,j}(x; \kappa) \equiv \mathcal{F}^{(\beta)}(x; (\mu_{(1)}, \dots, \mu_{(j-1)}, \kappa, \mu_{(j+1)}, \dots, \mu_{(P)})). \quad (2.4)$$

²We may always transform our parameter dependent function such that the parameters reside in the hypercube $[-1, 1]^P$.

2.2 Polynomial Approximation

In this section we recall some results from polynomial interpolation theory. We first describe a general interpolation framework for which we state three hypotheses. These hypotheses are the key ingredients in the proof of our new convergence theory in Section 3.

Let $\Gamma = [-1, 1]$, and let $f : \Gamma \rightarrow \mathbb{R}$ be a sufficiently smooth function. We introduce $N + 1$ distinct interpolation nodes $y_{N,i} \in \Gamma$, $0 \leq i \leq N$, and $N + 1$ characteristic functions $\chi_{N,i}$, $0 \leq i \leq N$, that satisfy $\chi_{N,i}(y_{N,j}) = \delta_{i,j}$, $0 \leq i, j \leq N$. We finally introduce an interpolation operator I_N defined by $I_N f = \sum_{i=0}^N f(y_{N,i})\chi_{N,i}$. We may now formally state our three hypotheses.

Hypothesis 1. *The error in the derivative of the interpolant $I_N f$ satisfies*

$$|f'(x) - (I_N f)'(x)| \leq G_f(N), \quad \forall x \in \Gamma, \quad (2.5)$$

where for a given f the function $G_f : \mathbb{N} \rightarrow (0, \infty)$ with $G_f(N) \rightarrow 0$ as $N \rightarrow \infty$.

Hypothesis 2. *The characteristic functions $\chi_{N,i}$, $0 \leq i \leq N$, satisfy*

$$\sum_{i=0}^N |\chi'_{N,i}(x)| \leq D(N), \quad \forall x \in \Gamma, \quad (2.6)$$

where the function $D : \mathbb{N} \rightarrow (0, \infty)$ is fixed (for a given interpolation scheme) with $D(N) \rightarrow \infty$ as $N \rightarrow \infty$.

Hypothesis 3. *Let $\epsilon \in \mathbb{R}^+$, and consider the equation*

$$G_f(N) = D(N)\epsilon \quad (2.7)$$

for the unknown N as $\epsilon \rightarrow 0$. Equation (2.7) has a solution $N = N(\epsilon) \geq 0$ that satisfies

$$\epsilon D(N(\epsilon)) \rightarrow 0 \quad (2.8)$$

as $\epsilon \rightarrow 0$.

We next consider several interpolation schemes and in each case confirm the corresponding instantiations of our hypotheses under suitable regularity conditions. First, we assume $f \in C^2(\Gamma)$ and consider piecewise linear interpolation over equidistant interpolation nodes $y_{N,i} = (2i/N - 1) \in \Gamma$, $0 \leq i \leq N$. In this case the characteristic functions $\chi_{N,i}$ are continuous and piecewise linear

Approximation of Parametric Derivatives by the EIM

“hat functions” with support only on the interval $[y_{N,0}, y_{N,1}]$ for $i = 0$, only on the interval $[y_{N,i-1}, y_{N,i+1}]$ for $1 \leq i \leq N - 1$, and only on the interval $[y_{N,N-1}, y_{N,N}]$ for $i = N$. For piecewise linear interpolation Hypothesis 1 and Hypothesis 2 obtain for

$$G_f(N) = 2N^{-1} \|f''\|_{L^\infty(\Gamma)}, \quad (2.9)$$

$$D(N) = N, \quad (2.10)$$

respectively. In this case (2.6) in Hypothesis 2 obtains with equality. We include the proofs in Appendix A.1. It is straightforward to demonstrate Hypothesis 3: we note that

$$N^{-1} = N\epsilon \quad (2.11)$$

has the solution $N(\epsilon) = \epsilon^{-1/2}$ and that $\epsilon^{-1/2}\epsilon \rightarrow 0$ as $\epsilon \rightarrow 0$.

Next, we assume $f \in C^3(\Gamma)$ and consider piecewise quadratic interpolation over equidistant interpolation nodes $y_{N,i} = (2i/N - 1) \in \Gamma$, $0 \leq i \leq N$. We assume that N is even such that we may divide Γ into $N/2$ intervals $[y_{N,i}, y_{N,i+2}]$, for $i = 0, 2, 4, \dots, N - 2$. The characteristic functions are for $x \in [y_{N,i}, y_{N,i+2}]$ then given as

$$\chi_{N,i}(x) = \frac{(x - y_{N,i+1})(x - y_{N,i+2})}{2h^2}, \quad (2.12)$$

$$\chi_{N,i+1}(x) = \frac{(x - y_{N,i})(x - y_{N,i+2})}{-h^2}, \quad (2.13)$$

$$\chi_{N,i+2}(x) = \frac{(x - y_{N,i})(x - y_{N,i+1})}{2h^2}, \quad (2.14)$$

for $i = 0, 2, 4, \dots, N - 2$, where $h = 2/N = y_{N,j+1} - y_{N,j}$, $0 \leq j \leq N - 1$. For piecewise quadratic interpolation Hypothesis 1 and Hypothesis 2 obtain for

$$G_f(N) = \text{const} \cdot N^{-2} \|f'''\|_{L^\infty(\Gamma)}, \quad (2.15)$$

$$D(N) = \frac{5}{2}N, \quad (2.16)$$

respectively. We include the proofs in Appendix A.2. It is straightforward to demonstrate Hypothesis 3: we note that

$$N^{-2} = N\epsilon \quad (2.17)$$

has the solution $N(\epsilon) = \epsilon^{-1/3}$ and that $\epsilon^{-1/3}\epsilon \rightarrow 0$ as $\epsilon \rightarrow 0$.

Paper 7

Finally, we assume that f is analytic in Γ and consider standard Chebyshev interpolation over the usual Chebyshev-nodes $y_{N,i} = -\cos(i\pi/N)$, $0 \leq i \leq N$. The characteristic functions are in this case the Lagrange polynomials $\chi_{N,i} \in \mathbb{P}_N(\Gamma)$ that satisfy $\chi_{N,i}(y_{N,j}) = \delta_{ij}$, $0 \leq i, j \leq N$. For Chebyshev interpolation Hypothesis 1 and Hypothesis 2 obtain for

$$G_f(N) = c_f N e^{-N \log(\rho_f)}, \quad (2.18)$$

$$D(N) = N^2, \quad (2.19)$$

respectively, where $c_f > 0$ and $\rho_f > 1$ depend only on f . In this case (2.6) in Hypothesis 2 obtains with equality. We refer to Reddy and Weideman [8] for a proof of (2.18) and to Rivlin [9, pp. 119–121] for a proof of (2.19). We finally demonstrate Hypothesis 3: we let $\eta = \log(\rho_f) > 0$ and we note that the transcendental equation

$$N e^{-N\eta} = N^2 \epsilon. \quad (2.20)$$

admits the solution

$$N(\epsilon) = \frac{1}{\eta} \mathcal{W}\left(\frac{\eta}{\epsilon}\right), \quad (2.21)$$

where \mathcal{W} denotes the Lambert W function(s) defined by $\xi = \mathcal{W}(\xi) e^{\mathcal{W}(\xi)}$ for any $\xi \in \mathbb{C}$. As $\xi \rightarrow \infty$, $\xi \in \mathbb{R}$, it can be shown [2] that $\mathcal{W}(\xi) < \log(\xi)$. Thus, as $\epsilon \rightarrow 0$, we obtain

$$N(\epsilon) < \frac{1}{\eta} \log\left(\frac{\eta}{\epsilon}\right) = \frac{1}{\eta} \left(\log(\eta) + \log(1/\epsilon) \right) \leq A \log(1/\epsilon) \quad (2.22)$$

for some sufficiently large constant A . We now consider the product $\epsilon(N(\epsilon))^2$ as $\epsilon \rightarrow 0$. By application of L'Hôpital's rule twice (Eqs. (2.25) and (2.27) below)

we obtain

$$\lim_{\epsilon \rightarrow 0} \epsilon(N(\epsilon))^2 \leq A^2 \lim_{\epsilon \rightarrow 0} \epsilon(\log(1/\epsilon))^2 \quad (2.23)$$

$$= A^2 \lim_{\epsilon \rightarrow 0} \frac{(\log(\epsilon))^2}{1/\epsilon} \quad (2.24)$$

$$= A^2 \lim_{\epsilon \rightarrow 0} \frac{2 \log(\epsilon)/\epsilon}{-1/\epsilon^2} \quad (2.25)$$

$$= 2A^2 \lim_{\epsilon \rightarrow 0} \frac{\log(\epsilon)}{-1/\epsilon} \quad (2.26)$$

$$= 2A^2 \lim_{\epsilon \rightarrow 0} \frac{1/\epsilon}{1/\epsilon^2} \quad (2.27)$$

$$= 2A^2 \lim_{\epsilon \rightarrow 0} \epsilon = 0. \quad (2.28)$$

Hypothesis 3 thus holds.

3 A General Convergence Result

We introduce an approximation space $W_M \equiv W_M(\Omega)$ of finite dimension M . For any $\mu \in \mathcal{D}$, our approximation to the function $\mathcal{F}(\cdot; \mu) : \Omega \rightarrow \mathbb{R}$ shall reside in W_M ; the particular approximation procedure invoked is not relevant for our theoretical results in this section. We show here that if, for any $\mu \in \mathcal{D}$, the error in the best $L^\infty(\Omega)$ approximation to $\mathcal{F}(\cdot; \mu)$ in W_M goes to zero as $M \rightarrow \infty$, then, for any multi-index β , $|\beta| \geq 0$, the error in the best $L^\infty(\Omega)$ approximation to $\mathcal{F}^{(\beta)}(\cdot; \mu)$ in W_M also goes to zero as $M \rightarrow \infty$. Of course, only modest M are of interest in practice: the computational cost associated with the approximation is M -dependent. However, our theoretical results in this section provide some promise that we may in practice invoke the “original” approximation space and approximation procedure also for the approximation of parametric derivatives.

We introduce, for any fixed $p \geq 0$ and any $M \geq 1$,

$$e_M^p \equiv \max_{\beta \in \mathcal{M}_p^P} \max_{\mu \in \mathcal{D}} \inf_{w \in W_M} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)}. \quad (3.1)$$

We then recall the definition of $\mathcal{J}_{\beta,j}$ from (2.4), and state

Proposition 1. *Let p be a fixed non-negative integer. Assume that Hypotheses 1, 2, and 3 hold for $f = \mathcal{J}_{\beta,j}(x; \cdot)$, $1 \leq j \leq P$, for all $x \in \Omega$, and for all*

Paper 7

$\beta \in \mathcal{M}_p^P$. In this case, if $e_M^p \rightarrow 0$ as $M \rightarrow \infty$, then

$$e_M^{p+1} \rightarrow 0 \quad (3.2)$$

as $M \rightarrow \infty$.

Proof. For each $x \in \Omega$, we introduce the interpolant $\mathcal{J}_{N,\beta,j}(x; \cdot) \equiv I_N \mathcal{J}_{\beta,j}(x; \cdot) \in \mathbb{P}_N(\mathcal{S})$ given by

$$\mathcal{J}_{N,\beta,j}(x; \cdot) \equiv I_N \mathcal{J}_{\beta,j}(x; \cdot) = \sum_{i=0}^N \mathcal{J}_{\beta,j}(x; y_{N,i}) \chi_{N,i}(\cdot); \quad (3.3)$$

recall that here, $\chi_{N,i} : \mathcal{S} \rightarrow \mathbb{R}$, $0 \leq i \leq N$, are characteristic functions that satisfy $\chi_{N,i}(y_{N,j}) = \delta_{i,j}$, $0 \leq i, j \leq N$.

Let $'$ denote differentiation with respect to the variable κ in (2.4). For each $x \in \Omega$ we consider an approximation to $\mathcal{J}'_{\beta,j}(x; \cdot)$ which we write as $\sum_{i=0}^N \chi'_{N,i} w_i(x)$, where $w_i \in W_M$, $1 \leq i \leq N$. We note that $\sum_{i=0}^N \chi'_{N,i}(\kappa) w_i \in W_M$ for all $\kappa \in \mathcal{S}$ (we define the w_i shortly; however we note that in general $\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) \notin W_M$, $1 \leq i \leq N$, are not valid choices). For the error in this approximation we note by the triangle inequality that (for any $w_i \in W_M$, $1 \leq i \leq N$)

$$\begin{aligned} \left\| \mathcal{J}'_{\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_i \right\|_{L^\infty(\Omega \times \mathcal{S})} &= \left\| \mathcal{J}'_{N,\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_i + \mathcal{J}'_{\beta,j} - \mathcal{J}'_{N,\beta,j} \right\|_{L^\infty(\Omega \times \mathcal{S})} \\ &\leq \left\| \mathcal{J}'_{N,\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_i \right\|_{L^\infty(\Omega \times \mathcal{S})} + \left\| \mathcal{J}'_{\beta,j} - \mathcal{J}'_{N,\beta,j} \right\|_{L^\infty(\Omega \times \mathcal{S})}. \end{aligned} \quad (3.4)$$

Here, $\mathcal{J}'_{N,\beta,j} \equiv (\mathcal{J}_{N,\beta,j})' = \sum_{i=0}^N \mathcal{J}_{\beta,j}(\cdot; y_{N,i}) \chi'_{N,i}(\cdot)$.

In our approximation, we use as coefficient functions $\chi'_{N,i}$ (and not, for example, $\chi_{N,i}$). With this choice and the definition of $\mathcal{J}'_{\beta,j}$, we may relate the error in our approximation to the error in the approximation of $\mathcal{J}_{\beta,j}$, which is our ultimate goal. For the first term on the right hand side of (3.4) we first

invoke (3.3), then the triangle inequality, and finally Hypothesis 2 to obtain

$$\begin{aligned}
 \left\| \mathcal{J}'_{N,\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_i \right\|_{L^\infty(\Omega \times \mathcal{S})} &= \left\| \sum_{i=0}^N (\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) - w_i) \chi'_{N,i} \right\|_{L^\infty(\Omega \times \mathcal{S})} \\
 &\leq \left\| \sum_{i=0}^N |\chi'_{N,i}| \|\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) - w_i\| \right\|_{L^\infty(\Omega \times \mathcal{S})} \\
 &\leq \left\| \max_{0 \leq i \leq N} |\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) - w_i| \sum_{j=0}^N |\chi'_{N,j}| \right\|_{L^\infty(\Omega \times \mathcal{S})} \\
 &\leq D(N) \max_{0 \leq i \leq N} \|\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) - w_i\|_{L^\infty(\Omega)}. \quad (3.5)
 \end{aligned}$$

Next, for any $\kappa \in \mathcal{S}$ we introduce the functions

$$w_{\beta,j}^*(\cdot; \kappa) \equiv \arg \inf_{w \in W_M} \|\mathcal{J}_{\beta,j}(\cdot; \kappa) - w\|_{L^\infty(\Omega)}. \quad (3.6)$$

We then consider (3.5) for $w_i = w_{\beta,j}^*(\cdot; y_{N,i})$ and note that

$$\begin{aligned}
 \left\| \mathcal{J}'_{N,\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_{\beta,j}^*(\cdot; y_{N,i}) \right\|_{L^\infty(\Omega \times \mathcal{S})} &\leq D(N) \max_{0 \leq i \leq N} \|\mathcal{J}_{\beta,j}(\cdot; y_{N,i}) - w_{\beta,j}^*(\cdot; y_{N,i})\|_{L^\infty(\Omega)} \\
 &\leq D(N) \max_{\kappa \in \mathcal{S}} \|\mathcal{J}_{\beta,j}(\cdot; \kappa) - w_{\beta,j}^*(\cdot; \kappa)\|_{L^\infty(\Omega)} \\
 &= D(N) \max_{\kappa \in \mathcal{S}} \inf_{w \in W_M} \|\mathcal{J}_{\beta,j}(\cdot; \kappa) - w\|_{L^\infty(\Omega)} \leq D(N) e_M^p, \quad (3.7)
 \end{aligned}$$

where the last step follows from the definition of e_M^p in (3.1).

For the second term on the right hand side of (3.4) we invoke Hypothesis 1 for $f = \tilde{f}_{\beta,j} \equiv \mathcal{J}_{\beta,j}(\tilde{x}_{\beta,j}; \cdot)$ to obtain

$$\|\mathcal{J}'_{\beta,j} - \mathcal{J}'_{N,\beta,j}\|_{L^\infty(\Omega \times \mathcal{S})} \leq G_{\tilde{f}_{\beta,j}}(N); \quad (3.8)$$

here $\tilde{x}_{\beta,j} \in \Omega$ is the particular point in Ω such that for given β and j , $\tilde{f}_{\beta,j}$ yields the “worst” behavior of the right-hand-side.

We now combine (3.4) for $w_i = w_{\beta,j}^*(\cdot; y_{N,i})$ with (3.7) and (3.8) to obtain

$$\left\| \mathcal{J}'_{\beta,j} - \sum_{i=0}^N \chi'_{N,i} w_{\beta,j}^*(\cdot; y_{N,i}) \right\|_{L^\infty(\Omega \times \mathcal{S})} \leq G_{\tilde{f}_{\beta,j}}(N) + D(N) e_M^p. \quad (3.9)$$

We then introduce $\beta_j^+ = \beta + e_j$ where e_j is the canonical unit vector with the j 'th entry equal to unity; we recall that β has length $|\beta| = p$ and hence β_j^+ has length $|\beta_j^+| = p + 1$. We note that the multi-index β , the parameter values $\mu_{(i)} \in \mathcal{D}_{(i)}$, $1 \leq i \leq P$, $i \neq j$, as well as the dimension j , were chosen arbitrarily above. We may thus conclude

$$\begin{aligned} \max_{\beta \in \mathcal{M}_p^P} \max_{1 \leq j \leq P} \max_{\mu \in \mathcal{D}} \left\| \mathcal{F}^{(\beta_j^+)}(\cdot; \mu) - \sum_{i=0}^N \chi'_{N,i}(\mu_{(j)}) w_{\beta,j}^*(\cdot; y_{N,i}) \right\|_{L^\infty(\Omega)} \\ \leq G_{\hat{f}}(N) + D(N) e_M^p \end{aligned} \quad (3.10)$$

(recall above we wrote $\kappa = \mu_{(j)}$ for each fixed j); here, $\hat{f} = \mathcal{J}_{\tilde{\beta}, \tilde{j}}(\tilde{x}_{\tilde{\beta}, \tilde{j}}; \cdot)$, where $1 \leq \tilde{j} \leq P$ and $\tilde{\beta} \in \mathcal{M}_p^P$ are the particular indices that yield the ‘‘worst’’ behavior of the right-hand-side.

We note that $\sum_{i=0}^N \chi'_{N,i}(\mu_{(j)}) w_{\beta,j}^*(\cdot; y_{N,i})$ is a particular member of W_M for any $\beta \in \mathcal{M}_p^P$, any $\mu_{(j)} \in \mathcal{D}_{(j)}$, and any $1 \leq j \leq P$. We thus obtain

$$e_M^{p+1} = \max_{\beta \in \mathcal{M}_{p+1}^P} \max_{\mu \in \mathcal{D}} \inf_{w \in W_M} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)} \leq G_{\hat{f}}(N) + D(N) e_M^p. \quad (3.11)$$

The final step is to bound the right-hand side of (3.11) in terms of e_M^p . To this end we note that we may choose N freely. In particular we may choose N as the minimizer $N = N_{\min}(e_M^p) > 0$ of the right hand side of (3.11); however for simplicity we shall make a different choice for N . Let $N_{\text{bal}}(e_M^p)$ denote the value of N that balances the two terms on the right hand side of (3.11); by Hypothesis 3 $N_{\text{bal}}(e_M^p)$ exists for sufficiently small e_M^p . We then choose $N = N_{\text{bal}}(e_M^p)$ in (3.11) to obtain

$$e_M^{p+1} \leq 2D(N_{\text{bal}}(e_M^p)) e_M^p, \quad (3.12)$$

where $e_M^{p+1} \rightarrow 0$ as $e_M^p \rightarrow 0$ by Hypothesis 3. \square

We now provide three lemmas. The first lemma quantifies the convergence in Proposition 1 in the case that $\mathcal{F}(x, \cdot) \in C^2(\mathcal{D})$ for all $x \in \Omega$.

Lemma 1. *Assume $\mathcal{F}(x, \cdot) \in C^2(\mathcal{D})$ for any $x \in \Omega$. If for any fixed $p \geq 0$ $e_M^p \rightarrow 0$ as $M \rightarrow \infty$, then there is a constant $C_{p+1} > 0$ such that*

$$e_M^{p+1} \leq C_{p+1} \sqrt{e_M^p} \quad (3.13)$$

as $M \rightarrow \infty$.

Proof. In this case we may invoke piecewise linear interpolation as our interpolation system in the proof of Proposition 1. By (2.9) and (2.10) we obtain $N_{\text{bal}}(e_M^p) = (2\|\hat{f}''\|_{L^\infty(\Gamma)}/e_M^p)^{1/2}$ and hence (3.12) for $D(N) = N$ becomes $e_M^{p+1} \leq 2(2\|\hat{f}''\|_{L^\infty(\Gamma)}/e_M^p)^{1/2}e_M^p$. The result follows for $C_{p+1} = 2(2\|\hat{f}''\|_{L^\infty(\Gamma)})^{1/2}$. \square

The next lemma quantifies the convergence in Proposition 1 in the case that $\mathcal{F}(x, \cdot) \in C^3(\mathcal{D})$ for all $x \in \Omega$.

Lemma 2. *Assume $\mathcal{F}(x, \cdot) \in C^3(\mathcal{D})$ for any $x \in \Omega$. If for any fixed $p \geq 0$ $e_M^p \rightarrow 0$ as $M \rightarrow \infty$, then there is a constant $C_{p+1} > 0$ such that*

$$e_M^{p+1} \leq C_{p+1}(e_M^p)^{2/3} \quad (3.14)$$

as $M \rightarrow \infty$.

Proof. In this case we may invoke piecewise quadratic interpolation as our interpolation system in the proof of Proposition 1. By (2.15) and (2.16) we obtain, for a positive constant \tilde{c} , $N_{\text{bal}}(e_M^p) = \tilde{c}(\|\hat{f}'''\|_{L^\infty(\Gamma)}/e_M^p)^{1/3}$ and hence (3.12) for $D(N) = 5N/2$ becomes $e_M^{p+1} \leq 5\tilde{c}(\|\hat{f}'''\|_{L^\infty(\Gamma)}/e_M^p)^{1/3}e_M^p$. The result follows for $C_{p+1} = 5\tilde{c}\|\hat{f}'''\|_{L^\infty(\Gamma)}^{1/3}$. \square

We make the following remark concerning Lemma 1 and Lemma 2 in the case of algebraic convergence.

Remark 1. *Let $|\beta| = p$, and assume that $\mathcal{F}^{(\beta)}(x, \cdot) \in C^{q_p}(\mathcal{D})$, $q_p > 0$, for all $x \in \Omega$. Assume that $e_M^p \sim M^{-r_p}$, $r_p > 0$, as $M \rightarrow \infty$; here the convergence rate r_p typically depends on the regularity q_p . For $q_p = 2$ we may invoke Lemma 1 to obtain*

$$e_M^{p+1} \leq C_{p+1}(e_M^p)^{1/2} \sim M^{-r_p/2} \sim M^{1-r_p+(r_p/2-1)} \sim M^{1+(r_p/2-1)}e_M^p. \quad (3.15)$$

Similarly, for $q_p = 3$ we may invoke Lemma 2 to obtain

$$e_M^{p+1} \leq C_{p+1}(e_M^p)^{2/3} \sim M^{-2r_p/3} \sim M^{1-r_p+(r_p/3-1)} \sim M^{1+(r_p/3-1)}e_M^p. \quad (3.16)$$

More generally, with higher-regularity versions of Lemma 1 and Lemma 2, we expect for any $q_p > 0$ that

$$e_M^{p+1} \leq C_{p+1}(e_M^p)^{1-1/q_p} \sim M^{-r_p(1-1/q_p)} \sim M^{1-r_p+(r_p/q_p-1)} \sim M^{1+(r_p/q_p-1)}e_M^p. \quad (3.17)$$

for any $q_p > 0$. We shall comment on these estimates further in our discussion of numerical results in Section 5.

The third lemma quantifies the convergence in Proposition 1 in the case that $\mathcal{F}(x, \cdot)$ is analytic over \mathcal{D} .

Lemma 3. *Assume $\mathcal{F}(x, \cdot) : \mathcal{D} \rightarrow \mathbb{R}$ is analytic over \mathcal{D} for any $x \in \Omega$. If for any fixed $p \geq 0$ $e_M^p \rightarrow 0$ as $M \rightarrow \infty$, then there is a constant $C_{p+1} > 0$ such that*

$$e_M^{p+1} \leq C_{p+1} (\log(e_M^p))^2 e_M^p \quad (3.18)$$

as $M \rightarrow \infty$. In particular, if for some p

$$e_M^p \sim M^\sigma e^{-\gamma M} \quad (3.19)$$

as $M \rightarrow \infty$, where σ is a non-negative constant and γ is a positive constant, then there is a constant C_{p+1} such that

$$e_M^{p+1} \leq C_{p+1} M^{\sigma+2} e^{-\gamma M} \quad (3.20)$$

as $M \rightarrow \infty$.

Proof. In this case we may invoke Chebyshev interpolation as our interpolation system in the proof of Proposition 1. By (2.18), (2.19), and (2.22) we obtain $N_{\text{bal}}(e_M^p) < \hat{c} \log(1/e_M^p)$ for a sufficiently large constant \hat{c} . Hence, with $D(N) = N^2$ and (3.12), we obtain $e_M^{p+1} \leq 2\hat{c}^2 (\log(1/e_M^p))^2 e_M^p$. The result (3.18) follows for $C_{p+1} = 2\hat{c}^2$ since $(\log(1/e_M^p))^2 = (\log(e_M^p))^{-2}$. The result (3.20) follows under the additional assumption (3.19) since in this case there is a constant B such that

$$N_{\text{bal}}(e_M^p) \leq B \log\left(\frac{1}{M^\sigma e^{-\gamma M}}\right) = B(-\sigma \log M + \gamma M) < B\gamma M, \quad (3.21)$$

and $D(N_{\text{bal}}(e_M^p)) \leq (B\gamma M)^2$. \square

Remark 2. *Note that, in Lemma 3, we can not obtain an explicit expression for the convergence rate of derivatives of order larger than $p+1$ (by for example an induction argument) since the result (3.20) is not sharp; an asymptotic lower bound for e_M^{p+1} is required to explicitly bound $N_{\text{bal}}(e_M^{p+1})$ as $M \rightarrow \infty$. Hence, we invoke an exact asymptotic relation in the assumption (3.19) in order to bound the convergence of the “next” derivative approximation based on the “current” derivative approximation.*

We also note that if the bound (3.20) were sharp, we could invoke the argument recursively to obtain an estimate of the form $e_M^p \sim M^{\sigma+2p} e^{-\gamma M}$.

4 The Empirical Interpolation Method

In this section we first recall the empirical interpolation method (EIM) [1, 5, 6] and then consider the convergence theory of the previous section applied to the EIM. The EIM approximation space is spanned by precomputed snapshots of a parameter dependent “generating function” for judiciously chosen parameter values from a predefined parameter domain. Given any new parameter value in this parameter domain, we can construct an approximation to the generating function at this new parameter value — or in fact an approximation to any function defined over the same spatial domain — as a linear combination of the EIM basis functions. The particular linear combination is determined through interpolation at judiciously chosen points in the spatial domain. For parametrically smooth functions, the EIM approximation to the generating function yields rapid, typically exponential, convergence.

4.1 Procedure

We introduce the generating function $\mathcal{G} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ such that for all $\mu \in \mathcal{D}$, $\mathcal{G}(\cdot; \mu) \in L^\infty(\Omega)$. We introduce a training set $\Xi_{\text{train}} \subset \mathcal{D}$ of finite cardinality $|\Xi_{\text{train}}|$ which shall serve as our computational surrogate for \mathcal{D} . We also introduce a triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$ of Ω with \mathcal{N} vertices over which we shall in practice, for any $\mu \in \mathcal{D}$, realize $\mathcal{G}(\cdot; \mu)$ as a piecewise linear function.

Now, for $1 \leq M \leq M_{\text{max}} < \infty$, we define the EIM approximation space $W_M^{\mathcal{G}}$ and the EIM interpolation nodes $T_M^{\mathcal{G}}$ associated with \mathcal{G} ; here, M_{max} is a specified maximum EIM approximation space dimension. We first choose (randomly, say) an initial parameter value $\mu_1 \in \mathcal{D}$; we then determine the first EIM interpolation node as $t_1 = \arg \sup_{x \in \Omega} |\mathcal{G}(x; \mu_1)|$; we next define the first EIM basis function as $q_1 = \mathcal{G}(\cdot; \mu_1) / \mathcal{G}(t_1; \mu_1)$. We can then, for $M = 1$, define $W_M^{\mathcal{G}} = \text{span}\{q_1\}$ and $T_M^{\mathcal{G}} = \{t_1\}$. We also define a nodal value matrix B^1 with (a single) element $B_{1,1}^1 = q_1(t_1) = 1$.

Next, for $2 \leq M \leq M_{\text{max}}$, we first compute the empirical interpolation of $\mathcal{G}(\cdot; \mu)$ for all $\mu \in \Xi_{\text{train}}$: we solve the linear system

$$\sum_{j=1}^{M-1} \phi_j^{M-1}(\mu) B_{i,j}^{M-1} = \mathcal{G}(t_i; \mu), \quad 1 \leq i \leq M-1, \quad (4.1)$$

and compute the empirical interpolation $\mathcal{G}_{M-1}(\cdot; \mu) \in W_{M-1}^{\mathcal{G}}$ as

$$\mathcal{G}_{M-1}(\cdot; \mu) = \sum_{i=1}^{M-1} \phi_i^{M-1}(\mu) q_i, \quad (4.2)$$

for all $\mu \in \Xi_{\text{train}}$. We then choose the next parameter $\mu_M \in \mathcal{D}$ as the maximizer of the EIM interpolation error over the training set,

$$\mu_M = \arg \max_{\mu \in \Xi_{\text{train}}} \|\mathcal{G}_{M-1}(\cdot; \mu) - \mathcal{G}(\cdot; \mu)\|_{L^\infty(\Omega)}; \quad (4.3)$$

note that thanks to our piecewise linear realization of $\mathcal{G}(\cdot; \mu)$, the norm evaluation is a simple comparison of function values at the \mathcal{N} vertices of $\mathcal{T}_{\mathcal{N}}(\Omega)$. We now choose the next EIM interpolation node as the point in Ω at which the EIM error associated with $\mathcal{G}_{M-1}(\mu_M)$ is largest,

$$t_M = \arg \sup_{x \in \Omega} |\mathcal{G}_{M-1}(x; \mu_M) - \mathcal{G}(x; \mu_M)|. \quad (4.4)$$

The next EIM basis function is then

$$q_M = \frac{\mathcal{G}_{M-1}(\cdot; \mu_M) - \mathcal{G}(\cdot; \mu_M)}{\mathcal{G}_{M-1}(t_M; \mu_M) - \mathcal{G}(t_M; \mu_M)}. \quad (4.5)$$

We finally enrich the EIM space: $W_M^{\mathcal{G}} = \text{span}\{q_1, \dots, q_M\}$; expand the set of nodes: $T_M^{\mathcal{G}} = \{t_1, \dots, t_M\}$; and expand the nodal value matrix: $B_{i,j}^M = q_j(t_i)$, $1 \leq i, j \leq M$.

Now, given any function $\mathcal{F} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ (in particular, we shall consider $\mathcal{F} = \mathcal{G}^{(\beta)}$), we define for any $\mu \in \mathcal{D}$ and for $1 \leq M \leq M_{\text{max}}$ the empirical interpolation of $\mathcal{F}(\cdot; \mu)$ in the space $W_M^{\mathcal{G}}$ (the space generated by \mathcal{G}) as

$$\mathcal{F}_M^{\mathcal{G}}(\cdot; \mu) = \sum_{i=1}^M \phi_i^M(\mu) q_i, \quad (4.6)$$

where the coefficients $\phi_i^M(\mu)$, $1 \leq i \leq M$, solve the linear system

$$\sum_{j=1}^M \phi_j^M(\mu) B_{i,j}^M = \mathcal{F}(t_i; \mu), \quad 1 \leq i \leq M. \quad (4.7)$$

We note that by construction the matrices $B^M \in \mathbb{R}^{M \times M}$, $1 \leq M \leq M_{\text{max}}$, are lower triangular: by (4.1), $\mathcal{G}_{M-1}(t_j; \mu_M) = \mathcal{G}(t_j; \mu_M)$ for $j < M$. As a

result, computation of the EIM coefficients ϕ_j^M , $1 \leq j \leq M$, in (4.7) and (4.1) are $\mathcal{O}(M^2)$ operations. We emphasize that the computational cost associated with the EIM approximation (4.6)–(4.7) (after snapshot precomputation), is independent of the number \mathcal{N} of vertices in the triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$. We may thus choose \mathcal{N} conservatively.

We next note that, for any multi-index β ,

$$(\mathcal{F}_M^{\mathcal{G}})^{(\beta)} = \left(\sum_{i=1}^M \phi_i^M(\mu) q_i \right)^{(\beta)} = \sum_{i=1}^M \varphi_i^M(\mu) q_i, \quad (4.8)$$

where $\varphi_i^M(\mu) = (\phi_i^M)^{(\beta)}(\mu)$, $1 \leq i \leq M$, solve the linear system (recall that the matrix B^M is μ -independent)

$$\sum_{j=1}^M \varphi_j^M(\mu) B_{i,j}^M = \mathcal{F}^{(\beta)}(t_i; \mu), \quad 1 \leq i \leq M. \quad (4.9)$$

Hence,

$$(\mathcal{F}_M^{\mathcal{G}})^{(\beta)} = (\mathcal{F}^{(\beta)})_M^{\mathcal{G}}, \quad (4.10)$$

that is, the parametric derivative of the approximation is equivalent to the approximation of the parametric derivative. We note that this equivalence holds since we invoke the same approximation space $W_M^{\mathcal{G}}$ for both EIM approximations $\mathcal{F}_M^{\mathcal{G}}$ and $(\mathcal{F}^{(\beta)})_M^{\mathcal{G}}$.

4.2 Convergence theory applied to the EIM

We introduce the Lebesgue constants [7]

$$\Lambda_M = \sup_{x \in \Omega} \sum_{i=1}^M |V_i^M(x)|, \quad 1 \leq M \leq M_{\max}, \quad (4.11)$$

where $V_i^M \in W_M^{\mathcal{G}}$ are the characteristic functions associated with $W_M^{\mathcal{G}}$ and $T_M^{\mathcal{G}}$: $V_i^M(t_j) = \delta_{ij}$, $1 \leq i, j \leq M$, where δ is the Kronecker delta symbol. It can be proven [1, 5] that the EIM error satisfies

$$\|\mathcal{F}(\cdot; \mu) - \mathcal{F}_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} \leq (1 + \Lambda_M) \inf_{w \in W_M^{\mathcal{G}}} \|\mathcal{F}(\cdot; \mu) - w\|_{L^\infty(\Omega)}, \quad (4.12)$$

for $1 \leq M \leq M_{\max}$. It can furthermore be proven that $\Lambda_M < 2^M - 1$; however, in actual practice the growth of Λ_M is much slower, as we shall observe below (see also results in [1, 5, 6]).

Our theory of Section 3 considers the convergence in the best approximation error. In the following remark we apply Lemma 3 within the context of the EIM.

Remark 3. *It can be shown [1, 5] that the error in the EIM derivative approximation satisfies*

$$\begin{aligned} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - (\mathcal{F}^{(\beta)})_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} \\ \leq (1 + \Lambda_M) \inf_{w \in W_M^{\mathcal{G}}} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)}, \end{aligned} \quad (4.13)$$

for any $\mu \in \mathcal{D}$ and any multi-index β . Assume that the best approximation error

$$e_M^p = \max_{\mu \in \mathcal{D}} \inf_{w \in W_M^{\mathcal{G}}} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)} \rightarrow 0 \quad (4.14)$$

as $M \rightarrow \infty$ for all $\mu \in \mathcal{D}$ and any multi-index β such that $|\beta| = p$ is a non-negative integer. We may then conclude from Lemma 3 and (4.13) that

$$\begin{aligned} \max_{\mu \in \mathcal{D}} \|\mathcal{F}^{(\beta')}(\cdot; \mu) - (\mathcal{F}^{(\beta')})_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} &\leq (1 + \Lambda_M) e_M^{p+1} \\ &\leq (1 + \Lambda_M) C_{p+1} (\log(e_M^p))^2 e_M^p, \end{aligned} \quad (4.15)$$

for any multi-index β' such that $|\beta'| = p + 1$.

The term $e_M^p \rightarrow 0$ as $M \rightarrow \infty$ by assumption and thus $e_M^{p+1} \rightarrow 0$ as $M \rightarrow \infty$ by Proposition 1. Hence, the convergence of the EIM derivative approximation associated with derivatives of order $p + 1$ depends on the growth of the Lebesgue constant; precisely, we must require

$$\Lambda_M e_M^{p+1} \rightarrow 0 \quad (4.16)$$

as $M \rightarrow \infty$. We recall that the Lebesgue constant typically grows only modestly and thus we expect in practice convergence of the EIM derivative approximation.

Clearly, if the EIM approximation associated with derivatives of order p converges, then the best approximation associated with derivatives of order p converges as well. Hence convergence of the EIM approximation associated with derivatives of order p implies convergence of the EIM approximation associated with derivatives of order $p + 1$ provided the Lebesgue constant grows sufficiently modestly.

We expect that Remark 1, Remark 2, and Remark 3 may be applied (non-rigorously) to EIM convergence if the growth of the Lebesgue constant is modest, since then the convergence rates associated with the best approximation and EIM approximation can not be very different.

For any $p \geq 0$ we introduce the maximum EIM error over $\mathcal{E} \subseteq \mathcal{D}$

$$\epsilon_{M,\max}^p(\mathcal{E}) \equiv \max_{\mu \in \mathcal{E}} \max_{\beta \in \mathcal{M}_p^P} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - (\mathcal{F}^{(\beta)})_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} \quad (4.17)$$

for $|\beta| = p$ and $1 \leq M \leq M_{\max}$. We also introduce a function $R_M : \mathcal{D} \rightarrow \mathbb{R}$ such that

$$\|\mathcal{F}^{(\beta)}(\cdot; \mu) - (\mathcal{F}^{(\beta)})_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} = R_M(\mu) \inf_{w \in W_M^{\mathcal{G}}} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)} \quad (4.18)$$

for $1 \leq M \leq M_{\max}$. We note that by (4.13) $1 \leq R_M(\mu) \leq 1 + \Lambda_M$ for all $\mu \in \mathcal{D}$. With (4.17) and (4.18) we then obtain, for any $p \geq 0$,

$$\begin{aligned} \epsilon_{M,\max}^p(\mathcal{E}) &= \max_{\mu \in \mathcal{E}} \max_{\beta \in \mathcal{M}_p^P} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - (\mathcal{F}^{(\beta)})_M^{\mathcal{G}}(\cdot; \mu)\|_{L^\infty(\Omega)} \\ &= \max_{\mu \in \mathcal{E}} \max_{\beta \in \mathcal{M}_p^P} \left(R_M(\mu) \inf_{w \in W_M^{\mathcal{G}}} \|\mathcal{F}^{(\beta)}(\cdot; \mu) - w\|_{L^\infty(\Omega)} \right) = R_M(\hat{\mu}_p) e_M^p \end{aligned} \quad (4.19)$$

for a particular $\hat{\mu}_p \in \mathcal{E}$. We now introduce the EIM error *degradation factor*

$$\rho_{M,p}(\mathcal{E}) \equiv \frac{\epsilon_{M,\max}^p(\mathcal{E})}{\epsilon_{M,\max}^0(\mathcal{E})}, \quad (4.20)$$

and note that

$$\rho_{M,p}(\mathcal{E}) = \frac{R_M(\hat{\mu}_p) e_M^p}{R_M(\hat{\mu}_0) e_M^0} \leq (1 + \Lambda_M) \frac{e_M^p}{e_M^0}. \quad (4.21)$$

We make two observations. First, the EIM error degradation factor is similar (for fixed p as a function of M) to the best approximation error degradation factor $\rho_{M,p}^* \equiv e_M^p/e_M^0$ if the Lebesgue constant grows slowly. Second, if the ratio $R_M(\hat{\mu}_p)/R_M(\hat{\mu}_0) \sim 1$ as $M \rightarrow \infty$, then $\rho_{M,p}(\mathcal{E})$ will be similar to $\rho_{M,p}^*$ independent of the Lebesgue constant.

In our discussion of each of our numerical examples in the next section we plausibly assume that the Lebesgue constant grows only modestly, and in

particular that $\rho_{M,p}(\mathcal{E})$ is similar to $\rho_{M,p}^*$. We confirm this assumption with explicit calculation of the Lebesgue constant.

The following remark is particularly relevant in our subsequent discussion of the sharpness of our theoretical results.

Remark 4. *Assume that the Lebesgue constant grows slowly and thus that the convergence rate associated with the EIM approximation is similar to the convergence rate associated with the best approximation. Consider the case of exponential convergence and assume that the bound provided by Lemma 3 is sharp. If $\epsilon_{M,\max}^0(\mathcal{D}) \sim M^\sigma e^{-\gamma M}$ for $\sigma, \gamma > 0$, we expect that $\epsilon_{M,\max}^p(\mathcal{D}) \sim M^{\sigma+2p} e^{-\gamma M}$, and thus an EIM error degradation factor $\rho_{M,p} \sim (M^{2p})$. As we shall observe shortly for our numerical results this estimate for the degradation factor is not quite sharp.*

We may obtain an expression for the EIM error degradation factor also in the case of algebraic convergence. However, the relation between the regularity of the function (q_p in Remark 1) and the convergence (r_p in Remark 1) is not a priori known for the EIM (or best) approximation. We thus save the discussion of the EIM error degradation factor in the case of algebraic convergence for our numerical results section, in which we compute the relation between q_p and r_p a posteriori.

5 Numerical Results

5.1 Example 1: Parametrically smooth Gaussian surface

We introduce the spatial domain $\Omega = [0, 1]^2$ and the parameter domain $\mathcal{D} = [0.4, 0.6]^2$. We consider the 2D Gaussian $\mathcal{F} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ defined by

$$\mathcal{F}(x; \mu) = \exp\left(\frac{-(x_{(1)} - \mu_{(1)})^2 - (x_{(2)} - \mu_{(2)})^2}{2\sigma^2}\right) \quad (5.1)$$

for $x \in \Omega$, $\mu \in \mathcal{D}$, and $\sigma \equiv 0.1$. This function is thus parametrized by the location of the maximum of the Gaussian surface. We note that for all $x \in \Omega$ the function $\mathcal{F}(x; \cdot) \in C^\infty(\mathcal{D})$; we may thus invoke Lemma 3.

We introduce a triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$ with $\mathcal{N} = 2601$ vertices; we introduce an equi-distant training set “grid” $\Xi_{\text{train}} \subset \mathcal{D}$ of size $|\Xi_{\text{train}}| = 900 = 30 \times 30$. We then pursue the EIM with $\mathcal{G} = \mathcal{F}$ for $M_{\max} = 130$.

We now introduce a uniformly distributed random test set $\Xi_{\text{test}} \subset \mathcal{D}$ of size 1000. In Figure 1 we show the maximum interpolation errors $\epsilon_{M,\max}^p(\Xi_{\text{test}})$

Approximation of Parametric Derivatives by the EIM

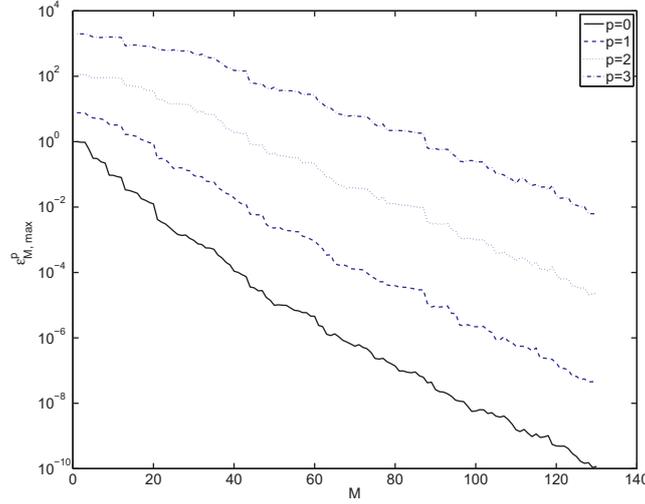


Figure 1: The maximum EIM error over the test set $\epsilon_{M,\max}^p(\Xi_{\text{test}})$ for $0 \leq p \leq 3$ for Example 1.

for $p = 0, 1, 2, 3$; the convergence is exponential (note the lin-log scaling of the axes). We note that for large M , the rate of convergence associated with the derivatives ($p > 1$) is close to the rate of convergence associated with the generating function ($p = 0$).

In Figure 2 we show the EIM error degradation factors $\rho_{M,p}(\Xi_{\text{test}})$ for $p = 1, 2, 3$ as functions of M . We observe that the degradation factors behave approximately as M^p : there is an M^p degradation of the convergence associated with the derivative approximation for $p > 0$ compared to the convergence associated with the original function.

From Remark 4 we recall that we would have expected $\rho_{M,p}(\Xi_{\text{test}}) \sim M^{2p}$ if our theoretical result (3.20) were sharp. Since in practice we observe $\rho_{M,p}(\Xi_{\text{test}}) \sim M^p$, we conclude that the result (3.20) is not in general sharp. We also note that the factor M^2 in (3.20) originates from the *sharp* result (2.19); hence with our present strategy for the proof of Proposition 1 it is not clear how to sharpen (3.20). However, we note that our theory captures the correct qualitative behavior: a degradation by an algebraic factor for the derivative approximation.

Finally, in Figure 3, we report the Lebesgue constant Λ_M . We note that the growth of the Lebesgue constant is only modest. The EIM derivative approxi-

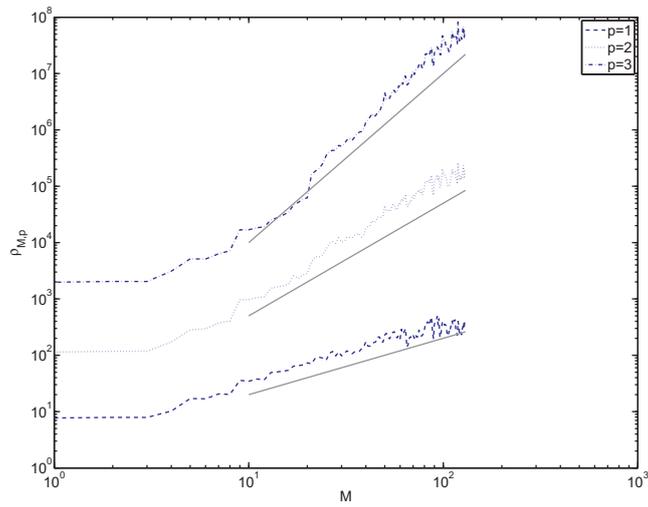


Figure 2: EIM error degradation factors $\rho_{M,p}(\Xi_{\text{test}})$, $p = 1, 2, 3$, for Example 1. The shorter solid gray lines represent exact rates M^p .

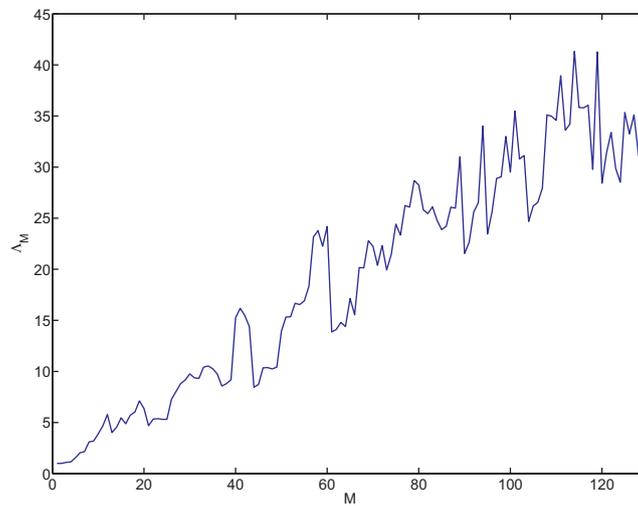


Figure 3: The Lebesgue constant Λ_M for Example 1.

Approximation of Parametric Derivatives by the EIM

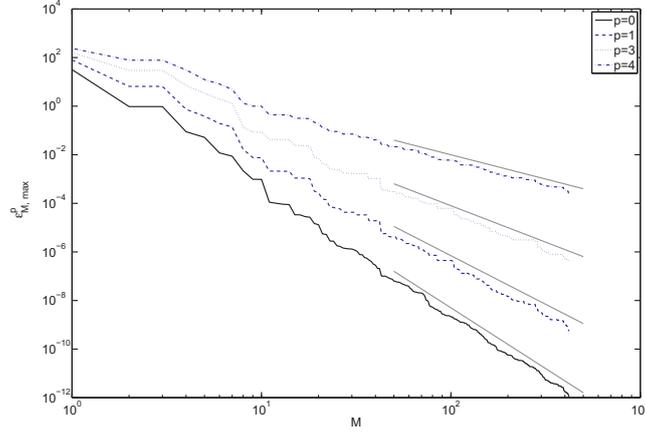


Figure 4: The maximum EIM error over the test set $\epsilon_{M, \max}^p(\Xi_{\text{test}})$ for $0 \leq p \leq 3$ for Example 2. The shorter gray lines represent exact rates M^{-5+p} .

mation is thus close to the best $L^\infty(\Omega)$ approximation in the space $W_M^{\mathcal{F}}$.

5.2 Example 2: A parametrically singular function

We introduce the spatial domain $\Omega = [-1, 1]$ and the parameter domain $\mathcal{D} = [-1, 1]$. We consider the function $\mathcal{F} : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ defined by

$$\mathcal{F}(x; \mu) = |x - \mu|^5 \quad (5.2)$$

for $x \in \Omega$ and $\mu \in \mathcal{D}$. The function thus has a singularity at $x = \mu$ for any $\mu \in \mathcal{D}$. For any $x \in \Omega$ we have $\mathcal{F}(x; \cdot) \in C^4(\mathcal{D})$. More generally, for any $x \in \Omega$ and $p = 0, 1, 2, 3$, we have $\mathcal{F}^{(p)}(x; \cdot) \in C^{q_p}(\mathcal{D})$ for $q_p = 4 - p$.

We introduce a triangulation $\mathcal{T}_{\mathcal{N}}(\Omega)$ with $\mathcal{N} = 1000$ vertices; we introduce an equi-distant training set “grid” $\Xi_{\text{train}} \subset \mathcal{D}$ of size $|\Xi_{\text{train}}| = 1000$. We then pursue the EIM with $\mathcal{G} = \mathcal{F}$ for $M_{\max} = 420$.

We now introduce a uniformly distributed random test set $\Xi_{\text{test}} \subset \mathcal{D}$ of size 1000. In Figure 4 we show the maximum interpolation errors $\epsilon_{M, \max}^p(\Xi_{\text{test}})$ for $p = 0, 1, 2, 3$; we observe for the convergence $\epsilon_{M, \max}^p(\Xi_{\text{test}}) \sim M^{-5+p}$. These results suggest that, in general, if $\mathcal{F}^{(p)} \in C^{q_p}(\mathcal{D})$, then $e_M^p \sim M^{-q_p-1}$, which corresponds to $r_p = q_p + 1$ in Remark 1.

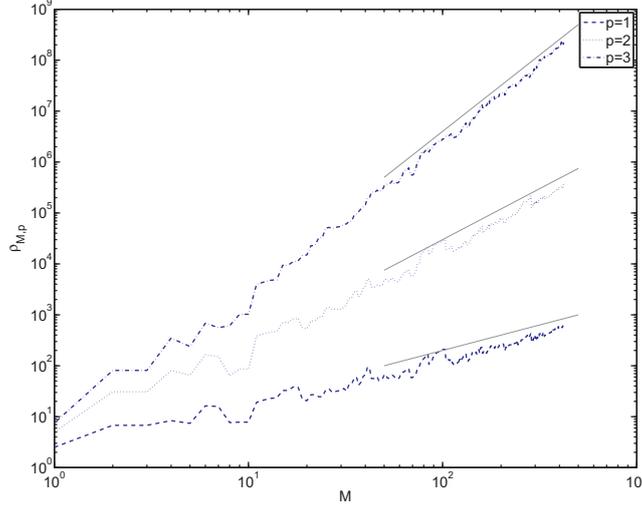


Figure 5: EIM error degradation factors $\rho_{M,p}(\Xi_{\text{test}})$, $p = 1, 2, 3$, for Example 2. The shorter solid gray lines represent exact rates M^p .

In Figure 5 we show the EIM error degradation factors $\rho_{M,p}(\Xi_{\text{test}})$ for $p = 1, 2, 3$ as functions of M . As for Example 1, we note that $\rho_{M,p}(\Xi_{\text{test}}) \sim M^p$ (of course this factor may in this case be interpreted directly from Figure 4).

With $r_p = q_p + 1$ in Remark 1, the estimate (3.17) in Remark 1 becomes

$$e_M^{p+1} \leq M^{1+\frac{1}{q_p}} e_M^p = M^{1+\frac{1}{4-p}} e_M^p. \quad (5.3)$$

If this is a sharp estimate, we expect for our example with $q_p = 4 - p$

$$e_M^1 \sim M^{1+\frac{1}{4}} e_M^0, \quad (5.4)$$

$$e_M^2 \sim M^{1+\frac{1}{3}} e_M^1 \sim M^{2+\frac{1}{3}+\frac{1}{4}} e_M^0, \quad (5.5)$$

$$e_M^3 \sim M^{1+\frac{1}{2}} e_M^2 \sim M^{3+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}} e_M^0. \quad (5.6)$$

From these estimates we may expect EIM error degradation factors

$$\rho_{M,p}(\Xi_{\text{train}}) \sim M^{p+\sum_{j=0}^{p-1} \frac{1}{4-j}}, \quad p = 1, 2, 3. \quad (5.7)$$

However, from our computations we see that this is not the case in practice: our results show $\rho_{M,p}(\Xi_{\text{test}}) \sim M^p$. We thus conclude that our theoretical results

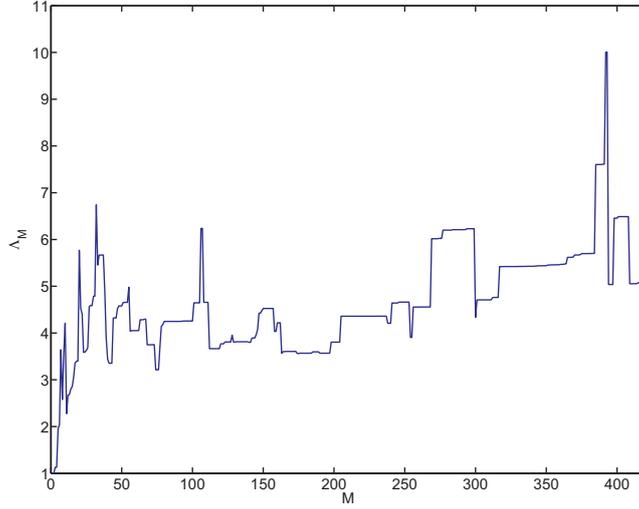


Figure 6: The Lebesgue constant Λ_M for Example 2.

in Lemma 1 and Lemma 2 (and higher order versions of these as indicated in Remark 1) are not sharp. The bounds predict $e_M^{p+1} \leq C_{p+1} M^{1+1/q_p} e_M^p$ for $\mathcal{F}^{(p)}(x; \cdot) \in C^{q_p}(\mathcal{D})$: a non-optimality of a factor M^{1/q_p} . We note that for functions with high regularity — large q_p — the sharpness of the bounds will improve since $e_M^{p+1} \leq M^{1+1/q_p} e_M^p \rightarrow M e_M^p$ as $q_p \rightarrow \infty$.

Finally, in Figure 6 we report the Lebesgue constant Λ_M ; any growth of the Lebesgue constant is hardly present. The EIM derivative approximation is thus close to the best $L^\infty(\Omega)$ approximation in the space $W_M^{\mathcal{F}}$.

6 Concluding remarks

We have introduced new *a priori* convergence theory for the approximation of parametric derivatives by a general approximation scheme. In particular, we have focused on approximation by the EIM both in our discussion and for our numerical results. The results suggest that the EIM may be invoked in practice for the approximation of parametric derivatives without construction of additional EIM spaces with the parametric derivatives as generating functions, or alternatively enrichment of the original space with parametric derivatives.

There are several opportunities for improvements of the theory. First, our numerical results suggest that it should be possible to sharpen the theoretical bounds. We note in our numerical results an EIM error degradation factor M^p for the convergence associated with the approximation of p 'th order derivatives for both parametrically analytic and parametrically non-analytic functions. In contrast, our theory and remarks predict a degradation factor M^{2p} for parametrically analytic functions, and a degradation factor $M^{p+\sum_{j=0}^{p-1} \frac{1}{s-j}}$ for parametrically non-analytic functions when the original function resides in $C^s(\mathcal{D})$ (but not in $C^{s+\alpha}(\mathcal{D})$ for arbitrarily small $\alpha > 0$).

Second, we would like to extend the validity of the theory to other (e.g. Sobolev) norms; in this case we may for example consider reduced basis [10] approximations to parametric derivatives of solutions to partial differential equations.

A Proofs for Hypotheses 1 and 2

A.1 Piecewise linear interpolation

We consider piecewise linear interpolation over the equidistant interpolation nodes $y_{N,i} = (2i/N - 1) \in \Gamma = [-1, 1]$, $0 \leq i \leq N$. In this case the characteristic functions $\chi_{N,i}$ are continuous and piecewise linear “hat functions” with support only on the interval $[y_{N,0}, y_{N,1}]$ for $i = 0$, on $[y_{N,i-1}, y_{N,i+1}]$ for $1 \leq i \leq N - 1$, and on $[y_{N,N-1}, y_{N,N}]$ for $i = N$.

We recall the results (2.9) and (2.10) from Section 2.2. Let $f : \Gamma \rightarrow \mathbb{R}$ with $f \in C^2(\Gamma)$. We then have, for any $x \in \Gamma$,

$$|f'(x) - (I_N f)'(x)| \leq 2N^{-1} \|f''\|_{L^\infty(\Gamma)} \quad (\text{A.1})$$

as $N \rightarrow \infty$. Further, for all $x \in \Gamma$, the characteristic functions $\chi_{N,i}$, $0 \leq i \leq N$, satisfy

$$\sum_{i=0}^N |\chi'_{N,i}(x)| = N. \quad (\text{A.2})$$

We first demonstrate (A.1) (and hence (2.9)). For $x \in [y_{N,i}, y_{N,i+1}]$, $0 \leq i \leq N - 1$, we have

$$(I_N f)'(x) = \frac{1}{h} (f(y_{N,i+1}) - f(y_{N,i})), \quad (\text{A.3})$$

Approximation of Parametric Derivatives by the EIM

where $h = 2/N$. We next write $f(y_{N,i})$ and $f(y_{N,i+1})$ as Taylor series around x as

$$f(y_{N,i}) = \sum_{j=0}^1 \frac{f^{(j)}(x)}{j!} (y_{N,i} - x)^j + \int_x^{y_{N,i}} f''(t)(y_{N,i} - t) dt, \quad (\text{A.4})$$

$$f(y_{N,i+1}) = \sum_{j=0}^1 \frac{f^{(j)}(x)}{j!} (y_{N,i+1} - x)^j + \int_x^{y_{N,i+1}} f''(t)(y_{N,i+1} - t) dt, \quad (\text{A.5})$$

which we then insert in the expression (A.3) for $(I_N f)'$ to obtain

$$\begin{aligned} (I_N f)'(x) - f'(x) &= \frac{1}{h} \int_x^{y_{N,i+1}} f''(t)(y_{N,i+1} - t) dt - \frac{1}{h} \int_x^{y_{N,i}} f''(t)(y_{N,i} - t) dt \\ &\leq \frac{1}{h} \|f''\|_{L^\infty(\Gamma)} \max_{x \in [y_{N,i}, y_{N,i+1}]} (|y_{N,i+1} - x|^2 + |y_{N,i} - x|^2) \\ &\leq h \|f''\|_{L^\infty(\Gamma)} = 2N^{-1} \|f''\|_{L^\infty(\Gamma)}. \end{aligned} \quad (\text{A.6})$$

We next demonstrate (A.2) (and hence (2.10)). It suffices to consider $x \in [y_{N,i}, y_{N,i+1}]$ for $0 \leq i \leq N-1$. On $[y_{N,i}, y_{N,i+1}]$ only $|\chi'_{N,i}(x)|$ and $|\chi'_{N,i+1}(x)|$ contribute to the sum; furthermore we have $|\chi'_{N,i}(x)| = |\chi'_{N,i+1}(x)| = 1/h = N/2$, from where the result (A.2) follows.

A.2 Piecewise quadratic interpolation

We consider piecewise quadratic interpolation over equidistant interpolation nodes $y_{N,i} = (2i/N - 1) \in \Gamma$, $0 \leq i \leq N$. We consider N equal such that we may divide Γ into $N/2$ intervals $[y_{N,i}, y_{N,i+2}]$, for $i = 0, 2, 4, \dots, N-2$. The characteristic functions $\chi_{N,i}$ are for $x \in [y_{N,i}, y_{N,i+2}]$ given as

$$\chi_{N,i}(x) = \frac{(x - y_{N,i+1})(x - y_{N,i+2})}{2h^2}, \quad (\text{A.7})$$

$$\chi_{N,i+1}(x) = \frac{(x - y_{N,i})(x - y_{N,i+2})}{-h^2}, \quad (\text{A.8})$$

$$\chi_{N,i+2}(x) = \frac{(x - y_{N,i})(x - y_{N,i+1})}{2h^2}, \quad (\text{A.9})$$

for $i = 0, 2, 4, \dots, N$.

We recall the results (2.15) and (2.16) from Section 2.2. Let $f : \Gamma \rightarrow \mathbb{R}$ with $f \in C^3(\Gamma)$. We then have, for any $x \in \Gamma$,

$$|f'(x) - (I_N f)'(x)| = \mathcal{O}(N^{-2}) \quad (\text{A.10})$$

Paper 7

as $N \rightarrow \infty$. Further, for all $x \in \Gamma$, the characteristic functions $\chi_{N,i}$, $0 \leq i \leq N$, satisfy

$$\sum_{i=0}^N |\chi'_{N,i}(x)| = \frac{5}{2}N. \quad (\text{A.11})$$

We first demonstrate (A.10). It suffices to consider the interpolant $I_N f(x)$ for $x \in [y_{N,i}, y_{N,i+2}]$, in which case

$$I_N f(x) = f(y_{N,i})\chi_{N,i}(x) + f(y_{N,i+1})\chi_{N,i+1}(x) + f(y_{N,i+2})\chi_{N,i+2}(x). \quad (\text{A.12})$$

Insertion of (A.7)–(A.9) and differentiation yields

$$\begin{aligned} (I_N f)'(x) &= \frac{1}{2h^2} \left(f(y_{N,i})(2x - y_{N,i+1} - y_{N,i+2}) \right. \\ &\quad \left. - 2f(y_{N,i+1})(2x - y_{N,i} - y_{N,i+2}) + f(y_{N,i+2})(2x - y_{N,i} - y_{N,i+1}) \right). \end{aligned} \quad (\text{A.13})$$

We next write $f(y_{N,i})$, $f(y_{N,i+1})$, and $f(y_{N,i+2})$ as Taylor series around x as

$$f(y_{N,i}) = \sum_{j=0}^3 \frac{f^{(j)}(x)}{j!} (y_{N,i} - x)^j + \mathcal{O}(h^4), \quad (\text{A.14})$$

$$f(y_{N,i+1}) = \sum_{j=0}^3 \frac{f^{(j)}(x)}{j!} (y_{N,i+1} - x)^j + \mathcal{O}(h^4), \quad (\text{A.15})$$

$$f(y_{N,i+2}) = \sum_{j=0}^3 \frac{f^{(j)}(x)}{j!} (y_{N,i+2} - x)^j + \mathcal{O}(h^4), \quad (\text{A.16})$$

where $h = 2/N = y_{N,j+1} - y_{N,j}$, $0 \leq j \leq N - 1$. We may then insert the expressions (A.14)–(A.16) into (A.13) to obtain

$$(I_N f)'(x) = f'(x) + \mathcal{O}(h^2). \quad (\text{A.17})$$

(For $j = 0$ and $j = 2$ the terms on the right-hand-side of (A.13) cancel. For $j = 1$ we obtain $f'(x)$ and for $j = 3$ we obtain $\mathcal{O}(h^2)$.)

We next demonstrate (A.11). It suffices to consider $x \in \Gamma_i \equiv [y_{N,i}, y_{N,i+2}]$. On Γ_i only $\chi'_{N,i}(x)$, $\chi'_{N,i+1}(x)$, and $\chi'_{N,i+2}(x)$ contribute to the sum. With

$h = 2/N = y_{j+1} - y_j$, $0 \leq j \leq N - 1$, we have

$$\max_{x \in \Gamma_i} |\chi'_{N,i}(x)| = \frac{N^2}{8} \max_{x \in \Gamma_i} |2x - y_{N,i+1} - y_{N,i+2}| = \frac{3}{4}N, \quad (\text{A.18})$$

$$\max_{x \in \Gamma_i} |\chi'_{N,i+1}(x)| = \frac{N^2}{4} \max_{x \in \Gamma_i} |2x - y_{N,i} - y_{N,i+2}| = N, \quad (\text{A.19})$$

$$\max_{x \in \Gamma_i} |\chi'_{N,i+2}(x)| = \frac{N^2}{8} \max_{x \in \Gamma_i} |2x - y_{N,i} - y_{N,i+1}| = \frac{3}{4}N. \quad (\text{A.20})$$

The result then follows.

Bibliography

- [1] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.
- [2] R. Corless, G. Gonnet, D. Hare, D. Jeffrey, and D. Knuth. On the LambertW function. *Advances in Computational Mathematics*, 5:329–359, 1996.
- [3] J. L. Eftang, M. A. Grepl, and A. T. Patera. A posteriori error bounds for the empirical interpolation method. *Comptes Rendus Mathematique*, 348(9-10):575 – 579, 2010.
- [4] M. Grepl. *A Posteriori Error Bounds for Reduced-Basis Approximations of Nonaffine and Nonlinear Parabolic Partial Differential Equations*. *Mathematical Models and Methods in Applied Sciences*, submitted, 2010.
- [5] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [6] Y. Maday, N. C. Nguyen, A. T. Patera, and G. S. H. Pau. A general multipurpose interpolation procedure: The magic points. *Communications in Pure and Applied Mathematics*, 8:383–404, 2009.
- [7] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics*, volume 37 of *Texts Appl. Math.* Springer, New York, 1991.

Paper 7

- [8] S. C. Reddy and J. A. C. Weideman. The accuracy of the Chebyshev differencing method for analytic functions. *SIAM Journal on Numerical Analysis*, 42(5):2176–2187 (electronic), 2005.
- [9] T. J. Rivlin. *The Chebyshev polynomials*. Wiley-Interscience [John Wiley & Sons], New York, 1974. Pure and Applied Mathematics.
- [10] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

