A Certified Reduced Basis Approach to
PDE-Constrained Optimization

by

Elizabeth Yi Qian

B.S. Aerospace Engineering (2014),
Massachusetts Institute of Technology

Submitted to the Department of Aeronautics & Astronautics
in partial fulfillment of the requirements for the degree of
Master of Science in Aerospace Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2017

© Massachusetts Institute of Technology 2017. All rights reserved.

Signature redacted

Author ............. Department of Aeronautics & Astronautics

January 26, 2017

Signature redacted

Certified by ...... Karen E. Willcox

Professor, Aeronautics & Astronautics

Thesis Supervisor

Signature redacted

Accepted by ............. Youssef M. Marzouk

Associate Professor, Aeronautics & Astronautics

Chair, Graduate Program Committee
DISCLAIMER NOTICE

Due to the condition of the original material, there are unavoidable flaws in this reproduction. We have made every effort possible to provide you with the best copy available.

Thank you.

The images contained in this document are of the best quality available.
A Certified Reduced Basis Approach to PDE-Constrained Optimization

by

Elizabeth Yi Qian

Submitted to the Department of Aeronautics & Astronautics on January 26, 2017, in partial fulfillment of the requirements for the degree of Master of Science in Aerospace Engineering

Abstract

Parameter optimization problems constrained by partial differential equations (PDEs) appear in many science and engineering applications. The PDE usually describes the underlying system or component behavior, while the parameters identify a particular configurations of the component, such as boundary and initial conditions, material properties, and geometry. Solving these optimization problems may require a prohibitively large number of computationally expensive PDE solves, particularly if the parameter dimension is high. It is therefore advantageous to replace expensive high-dimensional PDE solvers (e.g., finite element) with lower-dimension surrogate models.

This work builds on the reduced basis (RB) method, a model reduction method that allows efficient and reliable reduced order approximations for a large class of parametrized PDEs. Traditionally, RB models are generated during a computationally expensive offline phase for a certain admissible parameter space. The optimization problem can then be solved efficiently during the online phase. However, since the RB model is only evaluated along the optimization trajectory, building an RB model for the entire admissible parameter set incurs superfluous offline costs.

In this thesis, we break from the traditional RB offline/online decomposition and use a trust region framework to adaptively build the RB model along the optimization trajectory only. Novel a posteriori error bounds on the RB cost and cost gradient for quadratic cost functionals (e.g., least squares) are presented, and used to guarantee convergence to the optimum of the high-fidelity model. The proposed certified RB trust region approach uses high-fidelity solves to update the RB model only if the approximation is no longer sufficiently accurate, reducing the number of full-fidelity solves required. We consider problems governed by elliptic and parabolic PDEs and present numerical results for a thermal fin model problem in which we are able to reduce the number of full solves necessary for the optimization by up to 86%.

Thesis Supervisor: Karen E. Willcox
Title: Professor, Aeronautics & Astronautics
Acknowledgments

Thanks first to my family for their continuous support of my education and goals. Special thanks to David Darmofal, my undergraduate academic advisor, who always steered me right, and to Qiqi Wang, for introducing me to the ACDL and the wonderful field of computational science. Additional thanks to Kim Benard and Nora Delaney in the MIT Global Education & Career Development office, for their support and advice in writing the Fulbright grant application with which I was able to go to Germany to begin this research.

Thanks to Martin Grepl and Karen Veroy for welcoming me to Germany, and for their wisdom and guidance in this research and in life. Finally, a big thank you to my graduate advisor, Karen Willcox, for introducing me to Martin and Karen V. and supporting my Fulbright application, and for reliably excellent insight, perspective, and advice.

This project was supported in parts by the Fulbright U.S. Student Program, the National Science Foundation Graduate Research Fellowship Program, the Fannie and John Hertz Foundation, the Excellence Initiative of the German federal and state governments, the German Research Foundation through Grant GSC 111, and the US Department of Energy, Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics Program, awards DE-FG02-08ER2585 and DE-SC0009297, as part of the DiaMonD Multifaceted Mathematics Integrated Capability Center.
Contents

1 Introduction ................................................................. 13

2 Problem formulation ......................................................... 17
  2.1 Preliminaries ............................................................ 17
  2.2 Elliptic PDE-constrained optimization ................................ 20
  2.3 Parabolic PDE-constrained optimization ................................ 21

3 Reduced basis approximation and error estimation ......................... 23
  3.1 Reduced basis approximation and error estimation for elliptic problems 24
    3.1.1 Approximation ...................................................... 24
    3.1.2 A posteriori error estimation ...................................... 25
  3.2 Reduced basis approximation and error estimation for parabolic problems 29
    3.2.1 Approximation ...................................................... 29
    3.2.2 A posteriori error estimation ...................................... 30
  3.3 Computational procedure ................................................ 36
    3.3.1 Offline-online decomposition for reduced basis approximation . 37
    3.3.2 Offline-online decomposition for a posteriori error estimation . 40
    3.3.3 Parabolic case ..................................................... 44
    3.3.4 Summary of costs ................................................ 46

4 Trust region reduced basis algorithm ...................................... 47
  4.1 Trust region framework ................................................ 47
  4.2 Convergence ............................................................ 48
4.3 Algorithm ......................................................... 51

5 Numerical tests ................................................. 53
  5.1 Thermal fin model problem ................................. 54
    5.1.1 Elliptic model problem .............................. 54
    5.1.2 Parabolic model problem ............................ 55
    5.1.3 Problem data ........................................ 56
  5.2 Global reduced-basis approximation quality ............. 56
  5.3 Algorithm performance .................................... 59

6 Conclusions ...................................................... 63
List of Figures

5-1 Thermal fin geometry ......................................................... 54

5-2 Runtime comparison for optimizations constrained by elliptic PDEs. In contrast, the traditional offline-online reduced-basis approach for a 2D (6D) optimization runs in 0.04 (0.10) seconds online, but requires 1.6 (4800) seconds offline (on average). ......................................................... 60

5-3 Number of full model evaluations required for optimizations constrained by elliptic PDEs. The traditional offline-online RB approach requires 0 full evaluations online, and an average of 9 (48) full evaluations in the 2D (6D) case offline. ......................................................... 66

5-4 Runtime comparison for optimizations constrained by parabolic PDEs. For comparison, the average traditional offline-online RB 2D (6D) optimization runs in 0.23 (2.7) seconds online, but requires 125 seconds (60 hours) offline. ......................................................... 61

5-5 Number of full model evaluations required for optimizations constrained by parabolic PDEs. The traditional offline-online RB approach requires 0 full evaluations online, and an average of 15 (151) full evaluations in the 2D (6D) case offline. ......................................................... 61
# List of Tables

5.1 Trust region reduced basis algorithm parameters used in numerical tests 54
5.2 Problem data used for generation of numerical results. The $X$-inner product was defined as $(\cdot, \cdot)_X = a(\cdot, \cdot; \bar{\mu})$, for $\bar{\mu} = (1, 1, 1, 1, 1, 0.1)^T$ . . 56
5.3 2D elliptic thermal fin problem: convergence rate and effectivities of traditional reduced basis ................................. 58
5.4 6D elliptic thermal fin problem: convergence rate and effectivities of traditional reduced basis ........................................ 58
5.5 2D parabolic thermal fin problem: convergence rate and effectivities of traditional reduced basis ................................. 58
5.6 6D parabolic thermal fin problem: convergence rate and effectivities of traditional reduced basis ................................. 58
5.7 Summary of gains made by proposed trust region reduced basis optimization approach relative to MATLAB fmincon. Time gains are seen in the majority of test cases, and a reduction in the number of required FE function evaluations is seen in all cases tested, with the largest reduction in the 6D parabolic optimization. .................. 62
Chapter 1

Introduction

Optimization problems governed by partial differential equations (PDEs) appear in many settings across engineering and science disciplines, including engineering design optimization, optimal control problems, and inverse problems. Because typical optimization algorithms require numerous PDE evaluations, using classical discretization techniques (e.g., finite element) to solve these problems may be time-consuming, and in some cases, prohibitively expensive. One way to accelerate the solution of these problems is to replace expensive PDE evaluations with cheaper surrogate models. In this thesis, we consider surrogate models based on projection-based reduced models.

The use of surrogate models in optimization has an extensive literature (see, e.g., the review in [11]). Our interest is in formulations that retain convergence guarantees even when approximate information is employed throughout the optimization solution process. Trust region methods are one class of approaches that have a rich history of convergence results; see, for example, [9], for a detailed discussion of trust region methods. Traditionally, trust region methods replace high-fidelity objective function evaluations with local linear or quadratic Taylor expansions. These local approximations automatically satisfy first-order consistency conditions (that is, the approximate model’s objective and gradient evaluations are locally exact), which in turn provide guarantees that the resulting optimization solution will satisfy the optimality conditions of the original high-fidelity system. The influence of inexact gradient information is considered in [7, 26], and of inexact gradient and function information
in [6, 8, 9]. In [1], the authors consider a trust region framework with more general approximation models of varying fidelity and show how adaptive corrections may be used to achieve the first-order consistency conditions required to achieve a provably convergent formulation for general approximation models.

In addition to providing a theoretical framework that yields a convergent surrogate-based optimization formulation, trust region methods also provide an iterative framework for adaptation of the surrogate to the optimization problem of interest. Generating globally accurate surrogate models is typically prohibitively expensive, particularly when the underlying system is governed by PDEs. Thus, approaches that tailor the surrogate model to the optimization problem are of particular interest. In this thesis we consider a projection-based reduced model. While a number of adaptation approaches have been proposed for projection-based reduced models (see, e.g., [10, 18, 21, 23]), the challenge in the optimization setting is that regions of interest are not known a priori. Iterative approaches that adapt the reduced model as the optimization progresses have been considered in [5, 22]. We similarly adapt the reduced model as the optimization progresses, while also constructing our adaptation so as to rigorously address the convergence of the resulting optimization formulation.

We use the reduced basis (RB) method, a projection-based reduced-order modeling method, together with a trust region approach. The use of projection-based reduced models as surrogates in trust region optimization was first explored using proper orthogonal decomposition (POD) in [2]. In [2], the authors assume an upper bound on the inexactness of the function and gradient information resulting from the POD model and prove convergence of their algorithm using the results from [6, 26]. Unfortunately, verification of this upper bound in practice requires evaluation of the high-fidelity model. The work [30] further extends the results of [6, 26] to prove convergence (to a high-fidelity optimum) of a modified trust region algorithm that relies only on surrogate reduced-order model evaluations. In particular, [30] shows that by using upper bounds to the error in the cost functional and the gradient approximations, one can prove that the trust region solution converges to the exact solution. The work [30] further introduces heuristic error estimators for Krylov-Pade interpola-
tory reduced models and uses these estimators in the modified trust region algorithm. However, its reliance on heuristic estimators means that only heuristic convergence could be demonstrated and realized in practice. Heuristic error indicators have also been applied to trust region optimization for POD models [31], and in a stochastic context, an approximation based on sparse grids [17].

The reduced basis method is a reduced-order modeling technique for parametrized PDEs which supports rigorous a posteriori error estimation (see [25] for a review). We propose a reduced basis trust region method for solving optimization problems constrained by elliptic and parabolic PDEs which avoids the costly offline phase of the traditional RB method and iteratively builds the reduced model along the optimization trajectory as the algorithm progresses. After introducing the problem statement in Chapter 2, we present the following contributions:

1. In Chapter 3, we present a posteriori error bounds for the optimization cost functional and its gradient. Our bounds are based on a primal-dual formulation and are rigorous and efficiently computable. The dual formulation permits us to efficiently evaluate the gradient of the cost functional and at the same time derive error bounds for the cost functional which are superlinearly convergent with respect to the primal and dual error bound.

2. The error bounds play a crucial role in the reduced basis trust region method introduced in Chapter 4—they allow us to show convergence of the proposed approach to the (unknown) high-fidelity optimum. Furthermore, they allow us to efficiently control the accuracy of the reduced basis surrogate model during the optimization. We avoid the computationally expensive offline phase and build the reduced model adaptively along the optimization trajectory, thus keeping the number of high-fidelity solves to a minimum.

In Chapter 5, we present numerical results for parameter optimization problems constrained by elliptic and parabolic PDEs. We consider a thermal fin model problem with up to six variable parameters and compare the performance of our proposed reduced basis trust region approach to that of a traditional optimization using high-
fidelity PDE evaluations. We also compare to a “classical” RB approach, where the reduced model is first generated during an offline stage and then used for the optimization in the online stage.
Chapter 2

Problem formulation

In this chapter we introduce the PDE-constrained parameter optimization problem for both the elliptic and parabolic settings.

2.1 Preliminaries

Let \( \Omega \) be a physical domain in \( \mathbb{R}^d \) with Lipschitz continuous boundary \( \partial \Omega \). We define the Hilbert space \( X^e \) such that \( H^1(\Omega) \supset X^e \supset H^1_0(\Omega) \) and \( Y^e := L^2(\Omega) \), where
\[
H^1(\Omega) = \left\{ v \mid v \in L^2(\Omega), \nabla v \in (L^2(\Omega))^d \right\}, \quad H^1_0(\Omega) = \left\{ v \mid v \in H^1(\Omega), v|_{\partial \Omega} = 0 \right\},
\]
and \( L^2(\Omega) \) is the space of square-integrable functions over \( \Omega \). We associate with \( X^e \) and \( Y^e \) the inner products \( (w, v)_{X^e} \) and \( (w, v)_{Y^e} \) as well as the induced norms \( \|w\|_{X^e} = \sqrt{(w, w)_{X^e}} \) and \( \|w\|_{Y^e} = \sqrt{(w, w)_{Y^e}} \), respectively; for example, \( (w, v)_{X^e} := \int_{\Omega} \nabla w \cdot \nabla v + \int_{\Omega} w v, \forall w, v \in X^e \) and \( (w, v)_{Y^e} := \int_{\Omega} w v, \forall w, v \in Y^e \). We denote the corresponding dual spaces by \( X^{e'} \) and \( Y^{e'} \). The superscript \( .^e \) indicates that we are dealing with the "exact" continuous domain. Finally, let \( \mathcal{D} \subset \mathbb{R}^P \) be a \( P \)-dimensional set in which our \( P \)-tuple parameter \( \mu := (\mu_1, \ldots, \mu_P) \) resides.

We now define the conforming \( N \)-dimensional finite element (FE) approximation space \( X \subset X^e \) and define \( Y := Y^e \), inheriting inner product and norm definitions from \( X^e \) and \( Y^e \), respectively. For the parabolic case, we directly consider a time-discrete framework associated to the time interval \( I := [0, t_f] \), where \( I := [0, t_f] \) is divided into \( K \) uniform subintervals of length \( \Delta t = \frac{t_f}{K} \). We introduce \( \mathbb{K} := \{1, \ldots, K\} \) for
notational convenience, and define $t^k := k\Delta t, \forall k \in \mathbb{K}$, and finally, $I := \{t^0, \ldots, t^k\}$. We shall assume that $\mathcal{N}$ and $K$ are large enough (i.e. $X$ is sufficiently rich and the time-discretization sufficiently fine) that the FE approximation guarantees a desired accuracy over the whole parameter domain $D$.

We introduce the parameter-dependent bilinear form $a(\cdot, \cdot; \mu) : X \times X \to \mathbb{R}$ and its derivative in the $i$th component of $\mu$, $a_{\mu_i}(\cdot, \cdot; \mu) : X \times X \to \mathbb{R}, \forall i \in \{1, \ldots, P\}$. We also introduce the parameter-independent bilinear forms $m(\cdot, \cdot) : X \times X \to \mathbb{R}$ and $d(\cdot, \cdot) : X \times X \to \mathbb{R}$. We assume that all bilinear forms are continuous: for all $\mu \in D$,

\begin{equation}
0 < \gamma_a(\mu) := \sup_{w \in X \setminus \{0\}} \sup_{v \in X \setminus \{0\}} \frac{a(w, v; \mu)}{\|w\|_X \|v\|_X} \leq \gamma_0^a < \infty, \quad (2.1)
\end{equation}

\begin{equation}
0 < \gamma_{a_{\mu_i}}(\mu) := \sup_{w \in X \setminus \{0\}} \sup_{v \in X \setminus \{0\}} \frac{a_{\mu_i}(w, v; \mu)}{\|w\|_X \|v\|_X} \leq \gamma_0^{a_{\mu_i}} < \infty, \quad i = 1, \ldots, P \quad (2.2)
\end{equation}

\begin{equation}
0 < \gamma_m := \sup_{w \in X \setminus \{0\}} \sup_{v \in X \setminus \{0\}} \frac{m(w, v)}{\|w\|_Y \|v\|_Y} < \infty, \quad (2.3)
\end{equation}

\begin{equation}
0 < \gamma_d := \sup_{w \in X \setminus \{0\}} \sup_{v \in X \setminus \{0\}} \frac{d(w, v)}{\|w\|_X \|v\|_X} < \infty, \quad (2.4)
\end{equation}

and symmetric, i.e. $\forall w, v \in X, \forall \mu \in D$, $a(v, w; \mu) = a(w, v; \mu)$, $a_{\mu_i}(v, w; \mu) = a_{\mu_i}(w, v; \mu)$, $m(v, w) = m(w, v)$, and $d(v, w) = d(w, v)$. Additionally, we assume that $a(\cdot, \cdot; \mu)$ and $m(\cdot, \cdot)$ are coercive:

\begin{equation}
0 < \alpha_0^a := \inf_{v \in X} \frac{a(v, v; \mu)}{\|v\|_X^2}, \quad \forall \mu \in D, \quad 0 < \alpha_0^m := \inf_{v \in X} \frac{m(v, v)}{\|v\|_Y^2}. \quad (2.5)
\end{equation}

We next introduce two $X$-continuous linear functionals, the parameter-dependent $f(\cdot; \mu) : X \to \mathbb{R}$ and the parameter-independent $\ell(\cdot) : X \to \mathbb{R}$. Finally, we assume that all parameter-dependent linear and bilinear forms depend affinely on functions
of the parameter \( \mu \), i.e. we require that \( a(w, v; \mu) \) and \( f(v; \mu) \) can be expressed as

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

for all \( w, v \in X \) and \( \mu \in \mathcal{D} \), where \( Q_a \) and \( Q_f \) are some (preferably) small integers, the functions \( \Theta_a^q(\mu), \Theta_f^q(\mu) : \mathcal{D} \to \mathbb{R} \) are twice continuously differentiable and depend on \( \mu \), but the continuous bilinear and linear forms \( a^q(\cdot, \cdot) : X \times X \to \mathbb{R} \) and \( f^q : X \to \mathbb{R} \) do not depend on \( \mu \). We note that the functions \( \Theta_a^q(\mu) \) and \( \Theta_f^q(\mu) \) may depend nonlinearly on the parameter \( \mu \), and thus the forms \( a(w, v; \mu) \) and \( f(v; \mu) \) may also depend nonlinearly on \( \mu \). For simplicity, we assume that \( m(\cdot, \cdot) \), \( d(\cdot, \cdot) \), and \( \ell(\cdot) \) are parameter-independent, although extensions to affine parameter dependence are readily admitted [25]. We note that the bilinear and linear forms \( d \) and \( \ell \) will appear as the quadratic and linear cost terms in the next section, and that the bilinear form \( m \) represents the mass term in the parabolic problem statement.

For the development of the \textit{a posteriori} error bounds we also require the following ingredients. We assume that we have access to a positive lower bound \( \alpha_{LB}(\mu) : \mathcal{D} \to \mathbb{R}_+ \) for the coercivity constant \( \alpha(\mu) \) defined in Equation (2.5) such that

\[
0 < \alpha^a_0 \leq \alpha_{LB}(\mu) \leq \alpha(\mu), \quad \forall \mu \in \mathcal{D}, \tag{2.7}
\]

and an upper bound for the continuity constants \( \gamma_{a_{\mu_i}}(\mu) \) defined in Equation (2.2) such that

\[
\gamma_{a_{\mu_i}}^{UB}(\mu) \geq \gamma_{a_{\mu_i}}(\mu), \quad \forall \mu \in \mathcal{D}. \tag{2.8}
\]

We note that these lower and upper bounds are used in the \textit{a posteriori} error bound formulation to replace the actual coercivity and continuity constants, respectively. We thus require that these lower and upper bounds can be efficiently evaluated online, i.e., the computational cost is independent of the FE dimension \( N \). Various recipes exist to obtain such bounds [16, 25]; see Section 3.3 for more details.

19
2.2 Elliptic PDE-constrained optimization

We consider the constrained minimization of the output least-squares cost functional

\[
\min_{\mu \in \mathcal{D}} \| \mathcal{L}(u(\mu)) - g_{\text{ref}} \|^2_{\mathcal{D}} + \lambda \mathcal{R}(\mu)
\]

\[
\text{s.t. } u(\mu) \in X \text{ satisfies } a(u(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X,
\]

(2.9a, 2.9b)

where \( \mathcal{L} : X \to \mathcal{D} \) is a linear (output) functional and \( g_{\text{ref}} \in \mathcal{D} \) is a reference output, e.g. obtained from experimental measurements. Furthermore, \( \mathcal{D} \) is a suitable Hilbert space of observations with inner product \( (\cdot, \cdot)_{\mathcal{D}} \) and induced norm \( \| \cdot \|_{\mathcal{D}} = \sqrt{(\cdot, \cdot)_{\mathcal{D}}} \), and \( \mathcal{R} : \mathcal{D} \to \mathbb{R} \) continuous with \( \lambda \in \mathbb{R}_+ \) together form a scaled regularization term. Given our assumptions, it follows that the cost functional is continuous and thus at least one solution to Equation (2.9) exists [14]. We also note that constraints on the parameter can be directly handled using a barrier function approach; see e.g. [30].

Note that we may expand Equation (2.9a) with

\[
(\mathcal{L}(u) - g_{\text{ref}}, \mathcal{L}(u) - g_{\text{ref}})_{\mathcal{D}} = (\mathcal{L}(u), \mathcal{L}(u))_{\mathcal{D}} - 2(\mathcal{L}(u), g_{\text{ref}})_{\mathcal{D}} + (g_{\text{ref}}, g_{\text{ref}})_{\mathcal{D}}.
\]

Thus, defining \( d(w, v) := (\mathcal{L}(w), \mathcal{L}(v))_{\mathcal{D}}, \forall w, v \in X \) and \( \ell(v) := -2(\mathcal{L}(v), g_{\text{ref}})_{\mathcal{D}}, \forall v \in X \) and dropping the constant term \( (g_{\text{ref}}, g_{\text{ref}})_{\mathcal{D}} \), we obtain the following equivalent formulation for the optimization problem:

\[
\min_{\mu \in \mathcal{D}} J(\mu) \quad \text{where} \quad J(\mu) := d(u(\mu), u(\mu)) + \ell(u(\mu)) + \lambda \mathcal{R}(\mu)
\]

\[
\text{s.t. } u(\mu) \in X \text{ satisfies } a(u(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X.
\]

(2.10a, 2.10b)

In the sequel, we will use this more general quadratic cost formulation in developing the theory of the method.

Gradient-based optimization methods require access to the cost derivatives, which may be efficiently calculated using adjoint methods. We thus introduce the FE adjoint (dual) problem associated with our primal problem and cost in Equation (2.10) [27] as follows: Given \( \mu \in \mathcal{D} \) and the associated solution \( u(\mu) \) to Equation (2.10b), find
\( p(\mu) \in X \) satisfying
\[
a(v, p(\mu); \mu) = 2d(u(\mu), v) + \ell(v), \quad \forall v \in X. \tag{2.11}
\]

We also introduce the gradient of the cost function, \( \nabla_\mu J(\mu) \in \mathbb{R}^P \), with respect to the parameter \( \mu \) given by
\[
\nabla_\mu J(\mu) = \left( \frac{\partial J(\mu)}{\partial \mu_1}, \frac{\partial J(\mu)}{\partial \mu_2}, \ldots, \frac{\partial J(\mu)}{\partial \mu_P} \right)^T \tag{2.12}
\]

where \([14]\)
\[
\frac{\partial J(\mu)}{\partial \mu_i} = f_{\mu_i}(p(\mu); \mu) - a_{\mu_i}(u(\mu), p(\mu); \mu) + \lambda \frac{\partial \mathcal{R}(\mu)}{\partial \mu_i}, \quad i = 1, \ldots, P, \tag{2.13}
\]
is the partial derivative of the cost function with respect to the \( i \)-th parameter \( \mu_i \).

### 2.3 Parabolic PDE-constrained optimization

The parabolic optimization formulation is analogous to the elliptic case. We therefore directly consider the following (time-discrete) constrained minimization problem with quadratic cost:
\[
\min_{\mu \in \mathcal{D}} J(\mu) \quad \text{where} \quad J(\mu) := \Delta t \sum_{k=1}^{K} \left[ d(u^k(\mu), u^k(\mu)) + \ell(u^k(\mu)) \right] + \lambda \mathcal{R}(\mu) \tag{2.14a}
\]
s.t. \( u^k(\mu) \in X \) satisfies
\[
\frac{m(u^k(\mu) - u^{k-1}(\mu), v)}{\Delta t} + a(u^k(\mu), v; \mu) = f(v; \mu) y(t^k), \quad \forall v \in X, k \in \mathbb{K}, \tag{2.14b}
\]
with initial condition
\[
u^0(\mu) = 0, \tag{2.14c}
\]
where \( y(t^k) \) is a (known) time-dependent forcing input and we assume zero initial conditions for simplicity. Note that we consider an Euler-Backward discretization for the time integration and a time-independent regularization term; however, we
can also readily treat higher-order schemes such as Crank-Nicolson and also time-dependent regularization terms. Similar to the elliptic case, we introduce the FE adjoint (dual) problem associated with our primal problem Equation (2.14b) and cost in Equation (2.14a): Given \( \mu \in D \) and the associated solution \( u^k(\mu), \ k \in \mathbb{K} \) to Equation (2.14b), the adjoint \( p^k(\mu) \in X, K \geq k \geq 1 \), satisfies

\[
\frac{m(v, p^k(\mu) - p^{k+1}(\mu))}{\Delta t} + a(v, p^k(\mu); \mu) = 2d(u^k(\mu), v) + \ell(v), \quad \forall v \in X, k \in \mathbb{K},
\]

with final condition \( p^{K+1}(\mu) = 0 \). Note that the adjoint field variable evolves backward in time. Similar to the elliptic case, we define the gradient, \( \nabla_\mu J(\mu) \in \mathbb{R}^P \), with entries

\[
\frac{\partial J(\mu)}{\partial \mu_i} = \Delta t \sum_{k=1}^{K} \left[ f_{\mu_i}(p^k(\mu)) - a_{\mu_i}(u^k(\mu), p^k(\mu); \mu) \right] + \lambda \frac{\partial R(\mu)}{\partial \mu_i}, \quad i = 1, \ldots, P,
\]

which are the partial derivatives of the cost functional Equation (2.14a) with respect to the \( i \)th parameter \( \mu_i \).
Chapter 3

Reduced basis approximation and error estimation

The RB method is a projection-based model reduction method for parametrized PDEs [25]. Traditionally, it consists of an expensive, time-consuming offline phase, in which the reduced basis is built, and an inexpensive online phase, during which the pre-built RB may be exploited for rapid and certified simulations of the PDE at any parameter within the admissible parameter domain. In this section, we present primal-dual RB approximations and associated novel \textit{a posteriori} error estimation procedures for the elliptic and parabolic PDE-constrained parameter optimization problems introduced in the last section. To this end, we employ the RB approximations as surrogate models in the optimization problems Equation (2.10) and Equation (2.14) and develop new rigorous and efficiently evaluable error bounds for the cost functional and its gradient. In this work, we leverage these new error bounds to break from the offline/online paradigm in the optimization, i.e., we build the RB approximation during the iterative optimization procedure on-the-fly. Our error bounds guide the RB updates and at the same time allow us to guarantee convergence of the surrogate optimization to the (unknown) optimal solution of the original (FE) optimization problem. We note, however, that the results presented here also apply to the traditional offline/online RB setting.

Sections 3.1 and 3.2 present the RB approximation and error estimation results for
the elliptic and parabolic case, respectively. Section 3.3 discusses the computational aspects of the RB approximation.

### 3.1 Reduced basis approximation and error estimation for elliptic problems

This section introduces the RB approximation and error estimation results for the elliptic optimization problem (2.10).

#### 3.1.1 Approximation

Given X-orthogonal sets of primal and dual basis vectors $\zeta_n$ and $\psi_n$, $n = 1, \ldots, N$, we denote the $N$-dimensional primal and dual RB approximation spaces by $X_N^{pr}$ and $X_N^{du}$, defined as

$$X_N^{pr} := \text{span}\{\zeta_n, 1 \leq n \leq N\} = \text{span}\{u(\mu_n^{pr}), 1 \leq n \leq N\},$$

$$X_N^{du} := \text{span}\{\psi_n, 1 \leq n \leq N\} = \text{span}\{p(\mu_n^{du}), 1 \leq n \leq N\}.$$

We will comment on how $\mu_n^{pr}$ and $\mu_n^{du}$ are chosen in sections 3.3 and 4.2. For simplicity, we assume that the dimensions of the primal and dual RB spaces are the same, but results in this section extend directly to the case with different dimensions.

The RB approximation is then obtained via a Galerkin projection: Given $\mu \in \mathcal{D}$, the RB primal approximation $u_N(\mu) \in X_N^{pr}$ satisfies

$$a(u_N(\mu), v; \mu) = f(v; \mu), \quad \forall v \in X_N^{pr}, \quad (3.1)$$

and the RB dual approximation $p_N(\mu) \in X_N^{du}$ is given by

$$a(v, p_N(\mu); \mu) = 2d(u_N(\mu), v) + \ell(v), \quad \forall v \in X_N^{du}. \quad (3.2)$$
The RB cost functional and its derivative w.r.t. $\mu_i$ can be computed from

$$J_N(\mu) = d(u_N(\mu), u_N(\mu)) + \ell(u_N(\mu)) + \lambda R(\mu), \quad (3.3)$$

$$\frac{\partial J_N(\mu)}{\partial \mu_i} = f_{\mu_i}(p_N(\mu); \mu) - a_{\mu_i}(u_N(\mu), p_N(\mu); \mu) + \lambda \frac{\partial R(\mu)}{\partial \mu_i}. \quad (3.4)$$

### 3.1.2 A posteriori error estimation

We turn to the *a posteriori* error bounds. We first require the following definition:

**Definition 1.** The residuals of the primal and dual equations are defined by

$$r^{pr}(v; \mu) := f(v; \mu) - a(u_N(\mu), v; \mu), \quad \forall v \in X, \forall \mu \in D, \quad (3.5)$$

$$r^{du}(v; \mu) := 2d(u_N(\mu), v) + \ell(v) - a(v, p_N(\mu); \mu), \quad \forall v \in X, \forall \mu \in D. \quad (3.6)$$

We also define the primal and dual errors as follows:

$$e^{pr}(\mu) := u(\mu) - u_N(\mu) \quad \text{and} \quad e^{du}(\mu) := p(\mu) - p_N(\mu). \quad (3.7)$$

We can now prove the following lemma.

**Lemma 1.** Let $u(\mu)$ and $u_N(\mu)$ be the solutions to Equation (2.10b) and Equation (3.1), respectively. Furthermore, let $p(\mu)$ and $p_N(\mu)$ be the solutions to the associated dual equation Equation (2.11) and Equation (3.2). The error in the primal variable, $e^{pr}(\mu) = u(\mu) - u_N(\mu)$, is bounded by

$$\|e^{pr}(\mu)\|_X \leq \Delta^{pr}_N(\mu) := \frac{\|r^{pr}(\cdot; \mu)\|_{X'}}{\alpha_{LB}(\mu)}, \quad \forall \mu \in D, \quad (3.8)$$

and the error in the dual variable, $e^{du}(\mu) = p(\mu) - p_N(\mu)$, by

$$\|e^{du}(\mu)\|_X \leq \Delta^{du}_N(\mu) := \frac{\|r^{du}(\cdot; \mu)\|_{X'} + 2\gamma_d \Delta^{pr}_N(\mu)}{\alpha_{LB}(\mu)}, \quad \forall \mu \in D. \quad (3.9)$$
Proof. The bound Equation (3.8) is standard, see e.g. [25]. We follow an analogous procedure to show Equation (3.9). We first note from Equation (2.11) and Equation (3.6) that the dual error satisfies

\[ a(v, e_{du}(\mu); \mu) = r^{du}(v; \mu) + 2d(e_{pr}(\mu), v). \]  

(3.10)

Choosing \( v = e_{du}(\mu) \) and invoking Equation (2.5), Equation (2.7), and Equation (2.4) we obtain

\[ a_{LB}(\mu)\|e_{du}(\mu)\|^2_X \leq \|r^{du}(\cdot; \mu)\|_X \|e_{du}(\mu)\|_X + 2\gamma_d\|e_{pr}(\mu)\|_X \|e_{du}(\mu)\|_X. \]

The result Equation (3.9) then follows from Equation (3.8).

We may now consider the error in the cost functional and its gradient.

Theorem 1. The error in the cost functional, \( e^J(\mu) := J(\mu) - J_N(\mu) \), satisfies

\[ |e^J(\mu)| \leq \Delta^J_N(\mu) := \|r^{du}(\mu)\|_X \Delta^pr_N(\mu) + \gamma_d \Delta^pr_N(\mu)^2 + \|r^pr(p_N(\mu); \mu)|, \forall \mu \in D, \]  

(3.11)

where \( \Delta^pr_N(\mu) \) is the primal bound defined in Lemma 1.

Proof. It follows from Equation (2.10a) and Equation (3.3) that

\[ e^J(\mu) = d(\hat{u}(\mu), u(\mu)) - d(u_N(\mu), u_N(\mu)) + \ell(e_{pr}(\mu)). \]

Adding and subtracting \( r^pr(p_N(\mu); \mu) \) on the right-hand side and recalling the primal error-residual relationship, \( a(e_{pr}(\mu), v; \mu) = r^pr(v; \mu), \forall v \in X \), we obtain

\[ e^J(\mu) = d(u(\mu), u(\mu)) - d(u_N(\mu), u_N(\mu)) + \ell(e_{pr}(\mu)) + r^pr(p_N(\mu); \mu) - a(e_{pr}(\mu), p_N(\mu); \mu). \]

If we also add and subtract the term \( 2d(u_N(\mu), e_{pr}(\mu)) \) on the right-hand side and
invoke Equation (3.6), it follows that

\[ e^J(\mu) = r^d u(e^{pr}(\mu); \mu) + d(u(\mu), u(\mu)) - d(u_N(\mu), u_N(\mu)) + r^{pr}(p_N(\mu); \mu) - 2d(u_N(\mu), e^{pr}(\mu)). \]

Expanding \( d(u_N(\mu), e^{pr}(\mu)) = d(u_N(\mu), u(\mu) - u_N(\mu)) \), we obtain

\[ e^J(\mu) = r^d u(e^{pr}(\mu); \mu) + r^{pr}(p_N(\mu); \mu) + d(u(\mu), u(\mu)) - 2d(u_N(\mu), u(\mu)) + d(u_N(\mu), u_N(\mu)) \]

\[ = r^d u(e^{pr}(\mu); \mu) + r^{pr}(p_N(\mu); \mu) + d(e^{pr}(\mu), e^{pr}(\mu)). \]

Using the continuity of the bilinear form \( d \) yields

\[ |e^J(\mu)| \leq \|r^d u(\mu)\|_X \|e^{pr}(\mu)\|_X + \gamma_d \|e^{pr}(\mu)\|_X^2 + |r^{pr}(p_N(\mu); \mu)|. \]

The desired result directly follows from Lemma 1. \( \square \)

Before presenting the result for the cost gradient we make several remarks. First, since our goal is to develop effective \textit{a posteriori} error bounds for the cost functional as opposed to increasing the accuracy of the RB cost functional\(^1\), we incorporate the residual correction term in the bound Equation (3.11) instead of correcting the RB cost functional; see e.g. the discussion in [28]. Second, the dual problem plays two roles in our setting: it allows us to (i) efficiently compute the cost gradient from Equation (3.4) without having to resort to sensitivity derivatives; and (ii) devise an \textit{a posteriori} error bound for the cost functional which converges superlinearly to zero as \( N \to N \) with respect to the primal and dual bounds [24]. Finally, we note that certified reduced basis approximations for quadratic outputs have been previously considered in [15]. As opposed to the dual problem defined in Equation (2.11) in this thesis, the authors in [15] introduce a dual problem which is dependent on the RB solution \( u_N(\mu) \), e.g., for \( p(\mu) \in X \),

\[ a(v, p(\mu); \mu) = d(u(\mu) + u_N(\mu), v) + \ell(v), \quad \forall v \in X. \]

\(^1\)We will observe in Chapter 5 that the RB cost functional as defined in Equation (3.3) is sufficiently accurate for our purpose.
Although we would obtain a similar bound to Equation (3.11) for the cost functional using this formulation, the dual variable $p(\mu)$ thus defined cannot be used to compute the cost gradient from Equation (2.13). We now turn to the error bound for the cost gradient.

**Theorem 2.** The error in the cost gradient, $e^{\nabla J}(\mu) = \nabla \mu J(\mu) - \nabla \mu J_N(\mu)$, satisfies

$$\|e^{\nabla J}(\mu)\| \leq \Delta_N^{\nabla J}(\mu) := \left\|\Delta_N^{\nabla J}(\mu)\right\|, \quad (3.12)$$

where $\|\cdot\|$ is the Euclidean norm and $\Delta_N^{\nabla J}(\mu)$ is a vector whose $i$th component is the bound on the error in the $i$th component of the gradient, given by

$$\Delta_N^{\nabla J}(\mu) = \left\|f_{\mu_i}(\cdot; \mu)\right\|_{\mathcal{X}, \Delta_N^{du}(\mu)}$$

$$+ \gamma_{\mu_i}^{UB}(\mu) (\Delta_N^{pr}(\mu) \Delta_N^{du}(\mu) + \Delta_N^{pr}(\mu) \|p_N(\mu)\|_{\mathcal{X}} + \|u_N(\mu)\|_{\mathcal{X}} \Delta_N^{du}(\mu)), \quad (3.13)$$

where $\Delta_N^{pr}(\mu)$ and $\Delta_N^{du}(\mu)$ are the primal and dual error bounds defined in Lemma 1.

**Proof.** We consider the error in the derivative of the cost with respect to $\mu_i$, the $i$th element of the parameter vector $\mu$. It follows from Equation (2.13) and Equation (3.14) that

$$e^{\nabla_{\mu_i} J}(\mu) = f_{\mu_i}(e_{du}(\mu); \mu) - (a_{\mu_i}(u(\mu), p(\mu); \mu) - a_{\mu_i}(u_N(\mu), p_N(\mu); \mu)). \quad (3.14)$$

We next note that

$$a_{\mu_i}(u(\mu), p(\mu); \mu) - a_{\mu_i}(u_N(\mu), p_N(\mu); \mu) =$$

$$a_{\mu_i}(e_{pr}(\mu), e_{du}(\mu); \mu) + a_{\mu_i}(e_{pr}(\mu), p_N(\mu); \mu) + a_{\mu_i}(u_N(\mu), e_{du}(\mu); \mu). \quad (3.15)$$

Plugging Equation (3.15) into Equation (3.14) and invoking Equation (2.2) and Equation (2.8) we obtain

$$e^{\nabla_{\mu_i} J}(\mu) \leq \left\|f_{\mu_i}(\cdot; \mu)\right\|_{\mathcal{X}} \|e_{du}(\mu)\|_{\mathcal{X}} + \gamma_{\mu_i}^{UB}(\mu) \|e_{pr}(\mu)\|_{\mathcal{X}} \|e_{du}(\mu)\|_{\mathcal{X}}$$

$$+ \gamma_{\mu_i}^{UB}(\mu) \|e_{pr}(\mu)\|_{\mathcal{X}} \|p_N(\mu)\|_{\mathcal{X}} + \gamma_{\mu_i}^{UB}(\mu) \|u_N(\mu)\|_{\mathcal{X}} \|e_{du}(\mu)\|_{\mathcal{X}} \quad (3.16)$$
The result Equation (3.13) then follows from Lemma 1 and Equation (3.12) is obtained by taking the norm of all components.

In contrast to the \textit{a posteriori} error bound for the cost functional, the bound for the gradient does not exhibit a superlinear convergence with respect to the primal and dual error. However, even that fairly large relative errors (of 50\% or more) in the gradient are permissible in the trust-region framework without jeopardizing the overall convergence [6, 8]. We thus expect our gradient error bound to be sufficient to guarantee convergence of the RB trust region approach; see also the discussion in Section 4.2 and the numerical results in Chapter 5.

Although we consider only upper bounds for the various error terms in this thesis, lower bounds are sometimes of interest from a theoretical point of view, e.g. to quantify the convergence properties of greedy approaches to construct the reduced basis. Although it is possible to derive lower bounds for the error terms in Lemma 1 (see [25] for the primal error), similar results are not known for the error in the cost functional and gradient as well as for the parabolic case discussed in the next section. However, the convergence theory for the trust region method discussed in Chapter 4 only requires upper bounds, which is the main focus of the thesis.

### 3.2 Reduced basis approximation and error estimation for parabolic problems

This section introduces the RB approximation and develops the error estimation results for the parabolic case.

#### 3.2.1 Approximation

We introduce the primal and dual reduced basis spaces

\[
X^\text{pr}_N = \text{span}\{\zeta_n, 1 \leq n \leq N\}, \quad X^\text{du}_N = \text{span}\{\psi_n, 1 \leq n \leq N\},
\]
where the \( \zeta_n \) (and the \( \psi_n \), \( n = 1, \ldots, N \), are mutually \( X \)-orthogonal basis functions. We comment on their construction in sections 3.3 and 4.2.

The primal and dual RB approximations are obtained from a Galerkin projection: Given \( \mu \in \mathcal{D} \), the primal approximation \( u_N^k(\mu) \in X_N^{pr} \) to \( u^k(\mu) \in X \) satisfies

\[
\frac{m(u_N^k(\mu) - u_N^{k-1}(\mu), v)}{\Delta t} + a(u_N^k(\mu), v; \mu) = f(v; \mu)y(t^k), \quad \forall v \in X_N^{pr}, \tag{3.17}
\]

and the dual approximation \( p_N^k(\mu) \in X_N^{du} \) to \( p^k(\mu) \in X \) is given by

\[
\frac{m(v, p_N^k(\mu) - p_N^{k+1}(\mu))}{\Delta t} + a(v, p_N^k(\mu); \mu) = 2d(u_N^k(\mu), v) + \ell(v), \quad \forall v \in X_N^{du}. \tag{3.18}
\]

We can then calculate the RB cost and its derivative with respect to the \( \mu_i \) via

\[
J_N(\mu) := \Delta t \sum_{k=1}^K \left[ d(u_N^k(\mu), u_N^k(\mu)) + \ell(u_N^k(\mu)) \right] + \lambda \mathcal{R}(\mu), \tag{3.19}
\]

\[
\frac{\partial J_N(\mu)}{\partial \mu_i} = \Delta t \sum_{k=1}^K \left[ f_{u*, p_N^k(\mu)}(u_N^k(\mu) - p_N^k(\mu); \mu) \right] + \lambda \frac{\partial \mathcal{R}(\mu)}{\partial \mu_i}. \tag{3.20}
\]

### 3.2.2 A posteriori error estimation

The a posteriori error estimation procedure for the parabolic problem is analogous to that of the elliptic problem. In this section, we present the error bounds necessary for the trust region approach proposed in Chapter 4, deferring proofs to ??.. We first introduce the residuals in Definition 2.

**Definition 2.** The residuals of the primal and dual equations are defined by

\[
r_{pr}^k(v; \mu) = f(v)y(t^k) - a(u_N^k(\mu), v; \mu) - \frac{1}{\Delta t} m(u_N^k(\mu) - u_N^{k-1}(\mu), v), \tag{3.21}
\]

\[
r_{du}^k(v; \mu) = 2d(u_N^k(\mu), v) + \ell(v) - a(v, p_N^k(\mu); \mu) - \frac{1}{\Delta t} m(v, p_N^k(\mu) - p_N^{k+1}(\mu)), \tag{3.22}
\]

for all \( v \in X \) and all \( \mu \in \mathcal{D} \).
For the parabolic case, we also require the "spatio-temporal" energy norms for the primal and dual problem as follows.

**Definition 3.** The spatio-temporal energy norms for the primal and dual problem are given by

\[
|||v^k(\mu)|||_{pr} := \left[ m(v^k(\mu), v^k(\mu)) + \Delta t \sum_{k'=1}^{k} a(v^{k'}(\mu), v^{k'}(\mu); \mu) \right]^{\frac{1}{2}}, \quad \forall v \in X, \quad (3.23a)
\]

\[
|||v^k(\mu)|||_{du} := \left[ m(v^1(\mu), v^1(\mu)) + \Delta t \sum_{k'=k}^{K} a(v^{k'}(\mu), v^{k'}(\mu); \mu) \right]^{\frac{1}{2}}, \quad \forall v \in X. \quad (3.23b)
\]

We may now prove the following results for the primal and dual RB errors.

**Lemma 2.** Let \(u^k(\mu)\) and \(u^k_{N}(\mu), k \in \mathbb{K}\), be the solutions to Equation (2.14b) and Equation (3.17) and let \(p^k(\mu)\) and \(p^k_{N}(\mu), k \in \mathbb{K}\), be the solutions to the associated dual equations Equation (2.15) and Equation (3.18). Then, the following bounds for the error in the primal variable, \(e^k_{pr}(\mu) = u^k(\mu) - u^k_{N}(\mu)\), and the dual variable, \(e^k_{du}(\mu) = p^k(\mu) - p^k_{N}(\mu)\), hold for all \(\mu \in D\)

\[
|||e^k_{pr}(\mu)|||_{pr} \leq \Delta_{N,K}^{pr}(\mu) := \left( \frac{\Delta t}{\alpha_{LB}(\mu)} \sum_{k=1}^{K} \|r^k_{pr}(; \mu)\|_{X'}^2 \right)^{\frac{1}{2}}, \quad (3.24)
\]

\[
|||e^k_{du}|||_{du} \leq \Delta_{N,1}^{du}(\mu) := \left( 8\gamma^2 \left( \alpha_{pr}/\alpha_{LB}(\mu) \right)^2 + \frac{2\Delta t}{\alpha_{LB}(\mu)} \sum_{k=1}^{K} \|r^k_{du}(; \mu)\|_{X'}^2 \right)^{\frac{1}{2}}. \quad (3.25)
\]

**Proof.** We refer to [12] for the proof of eq. (3.24). We thus only need to show eq. (3.25). It follows from eq. (2.15) and eq. (3.22) that the error in the dual variable, \(e^k_{du} \in X\), satisfies

\[
\begin{align*}
m(v, e^k_{du} - e^{k+1}_{du}) + \Delta t \cdot a(v, e^k_{du}; \mu) &= \Delta t \cdot 2d\left(e^k_{pr}, v\right) + \Delta t \cdot r^k_{du}(v; \mu).
\end{align*}
\]
Following the usual procedure, we choose $v = e^k_{du}$ to obtain

$$m(e^k_{du}, e^{k+1}_{du}) + \Delta t \cdot a(e^k_{du}, e^k_{du}; \mu) = 2\Delta t \cdot d(e^k_{du}, e^k_{pr}) + \Delta t \cdot r^k_{du}(e^k_{du}; \mu). \quad (3.27)$$

First, we invoke the Cauchy-Schwarz and Young’s inequalities to get

$$m(e^k_{du}, e^{k+1}_{du}) \leq \frac{1}{2} m(e^k_{du}, e^k_{du}) + \frac{1}{2} m(e^{k+1}_{du}, e^{k+1}_{du}). \quad (3.28)$$

Also, from Young’s inequality it follows that

$$r^k_{du}(e^k_{du}(\mu); \mu) \leq \frac{1}{\alpha_{LB}} \| r^k_{du}(\cdot; \mu) \|_{X'}^2 + \frac{\alpha_{LB}}{4} \| e^k_{du}(\mu) \|_X^2, \quad (3.29)$$

and

$$d(e^k_{du}, e^k_{pr}) \leq \frac{2\gamma_d^2}{\alpha_{LB}(\mu)} \| e^k_{pr} \|_X^2 + \frac{\alpha_{LB}}{8} \| e^k_{du} \|_X^2; \quad (3.30)$$

where we also used the continuity of eq. (2.4) of the bilinear form $d$. Substituting eq. (3.28), eq. (3.29), and eq. (3.30) into eq. (3.27) we obtain

$$m(e^k_{du}, e^k_{du}) - m(e^{k+1}_{du}, e^{k+1}_{du}) + 2\Delta t \cdot a(e^k_{du}, e^k_{du}; \mu)$$

$$\leq \frac{2\Delta t}{\alpha_{LB}(\mu)} \| r^k_{du}(\cdot; \mu) \|_{X'}^2 + \frac{8\gamma_d^2 \Delta t}{\alpha_{LB}(\mu)} \| e^k_{pr} \|_X^2 + \alpha_{LB}(\mu) \Delta t \| e^k_{du} \|_X^2 \quad (3.31)$$

Finally, summing over $k = 1..K$ it follow that

$$m(e^1_{du}, e^1_{du}) + \Delta t \sum_{k=1}^K a(e^k_{du}(\mu), e^k_{du}(\mu); \mu) \leq$$

$$\sum_{k=1}^K \left( \frac{8\gamma_d^2 \Delta t}{\alpha_{LB}(\mu)} \| e^k_{pr} \|_X^2 + \frac{2\Delta t}{\alpha_{LB}(\mu)} \| r^k_{du}(\cdot; \mu) \|_{X'}^2 \right). \quad (3.32)$$

The result eq. (3.25) follows from eq. (3.24).

With Lemma 2 in hand, we may bound the parabolic cost and cost gradient as we did in Theorems 1 and 2 for the elliptic case.
Theorem 3. The error in the cost functional, \( e^J(\mu) := J(\mu) - J_N(\mu) \), may be bounded by

\[
e^J(\mu) \leq \Delta^J_N(\mu) := \left( \Delta t \sum_{k=1}^{K} \left\| r_{du}^k (\cdot; \mu) \right\|_{X'}^2 \right)^{1/2} \frac{\Delta^{pr}_{N,K}(\mu)}{\sqrt{\alpha_{LB}(\mu)}} + \frac{\gamma_d}{\alpha_{LB}(\mu)} (\Delta^{pr}_{N,K}(\mu))^2 + \Delta t \sum_{k=1}^{K} \left| \tau_{pr}^k (p_N^k(\mu); \mu) \right|, \ \forall \mu \in \mathcal{D}, \ (3.33)
\]

where \( \Delta^{pr}_{N,K}(\mu) \) is defined in Lemma 2.

Proof. We first note from eq. (2.14a) and eq. (3.19) that

\[
e^J(\mu) = \Delta t \sum_{k=1}^{K} \left\{ d(u^k(\mu), u^k(\mu)) - d(u_N^k(\mu), u_N^k(\mu)) + \ell(e^k_{pr}(\mu)) \right\} \quad (3.34)
\]

Adding and subtracting \( d(2u_N^k(\mu), e^k_{pr}(\mu)) \) within the sum, we obtain

\[
e^J(\mu) = \Delta t \sum_{k=1}^{K} \left\{ d(2u_N^k(\mu), e^k_{pr}(\mu)) + \ell(e^k_{pr}(\mu)) \right\} + d(u^k(\mu), u^k(\mu)) - d(u_N^k(\mu), u_N^k(\mu)) - d(2u_N^k(\mu), e^k_{pr}(\mu)) \quad (3.35)
\]

and note that

\[
d(u^k(\mu), u^k(\mu)) - d(u_N^k(\mu), u_N^k(\mu)) - d(2u_N^k(\mu), e^k_{pr}(\mu))
= d(u^k(\mu), u^k(\mu)) - d(2u_N^k(\mu), u^k(\mu)) + d(u_N^k(\mu), u_N^k(\mu))
= d(e^k_{pr}(\mu), e^k_{pr}(\mu)) \quad (3.36)
\]

Thus, combining eq. (3.35) and eq. (3.36), we have

\[
e^J(\mu) = \Delta t \sum_{k=1}^{K} \left\{ d(2u_N^k(\mu), e^k_{pr}(\mu)) + \ell(e^k_{pr}(\mu)) + d(e^k_{pr}(\mu), e^k_{pr}(\mu)) \right\}. \quad (3.37)
\]

Taking eq. (3.22) with \( v = e^k_{pr}(\mu) \), we can express the first two terms of eq. (3.37) as
follows:

\[
\sum_{k=1}^{K} \{d(2u^k_N(\mu), e^k_{pr}(\mu)) + \ell(e^k_{pr}(\mu))\} = \sum_{k=1}^{K} \{r^k_{du}(e^k_{pr}(\mu); \mu) + a(e^k_{pr}(\mu), p^k(\mu); \mu) \\
+ \frac{1}{\Delta t} m(e^k_{pr}(\mu), p^k_N(\mu) - p^{k+1}_N(\mu))\}.
\]

(3.38)

Since \(e^0_{pr}(\mu) = 0\) and \(p^{k+1}_N(\mu) = 0\), we observe that

\[
\sum_{k=1}^{K} m(e^k_{pr}(\mu), p^k_N(\mu) - p^{k+1}_N(\mu)) = \sum_{k=1}^{K} m(e^k_{pr}(\mu) - e^{k+1}_{pr}(\mu), p^k_N(\mu)).
\]

(3.39)

We may then substitute the primal error-residual relationship,

\[
r^k_{pr}(v; \mu) = a(e^k_{pr}(\mu), v; \mu) + \frac{1}{\Delta t} m(e^k_{pr}(\mu) - e^{k+1}_{pr}(\mu), v), \quad \forall v \in X,
\]

with \(v = p^k_N(\mu)\) into eq. (3.38) to obtain

\[
\sum_{k=1}^{K} \{d(2u^k_N(\mu), e^k_{pr}(\mu)) + \ell(e^k_{pr}(\mu))\} = \sum_{k=1}^{K} \{r^k_{du}(e^k_{pr}(\mu); \mu) + r^k_{pr}(p^k_N(\mu); \mu)\}.
\]

(3.40)

Substituting eq. (3.40) into eq. (3.37), we get

\[
e^f(\mu) = \Delta t \sum_{k=1}^{K} \{r^k_{du}(e^k_{pr}(\mu); \mu) + r^k_{pr}(p^k_N(\mu); \mu) + d(e^k_{pr}(\mu), e^k_{pr}(\mu))\}.
\]

(3.41)

It then follows from the continuity of \(d(\cdot, \cdot)\) that

\[
e^f(\mu) \leq \Delta t \sum_{k=1}^{K} \{\|r^k_{du}(\cdot; \mu)\|_X \|e^k_{pr}(\mu)\|_X + \gamma_d \|e^k_{pr}(\mu)\|^2_X + r^k_{pr}(p^k_N(\mu); \mu)\}.
\]

(3.42)

We now invoke the Cauchy-Schwarz inequality to the first term on the right hand
side which yields

\[
e^J(\mu) \leq \left( \Delta t \sum_{k=1}^{K} \| r_{du}^k (\cdot; \mu) \|_{X'}^2 \right)^{\frac{1}{2}} \left( \Delta t \sum_{k=1}^{K} \| e_{pr}^k (\mu) \|_{X}^2 \right)^{\frac{1}{2}} + \gamma_d \Delta t \sum_{k=1}^{K} \| e_{pr}^k (\mu) \|_{X}^2 + \Delta t \sum_{k=1}^{K} r_{pr}^k (p_N^k (\mu); \mu) \right). \tag{3.43}
\]

Finally, we note that

\[
\Delta t \sum_{k=1}^{K} \| e_{pr}^k (\mu) \|_{X}^2 \leq \frac{\Delta t}{\alpha_{LB}(\mu)} \sum_{k=1}^{K} a(e_{pr}^k (\mu), e_{pr}^k (\mu); \mu) \leq \frac{1}{\alpha_{LB}(\mu)} \| e_{pr}^k (\mu) \|_{pr} \tag{3.44}
\]

to arrive at the desired results by invoking eq. (3.24). \qed

**Theorem 4.** The error in the cost gradient, \( e_{\nabla J}(\mu) = \nabla_{\mu} J(\mu) - \nabla_{\mu} J_N(\mu) \), satisfies

\[
\| e_{\nabla J}(\mu) \| \leq \Delta_{N}^{\nabla J}(\mu) := \| \Delta_{N}^{\nabla_{\mu} J}(\mu) \|, \quad \forall \mu \in D, \tag{3.45}
\]

where \( \Delta_{N}^{\nabla_{\mu} J}(\mu) \) is a vector whose ith component is the bound on the error of the ith component of the gradient, given by

\[
\Delta_{N}^{\nabla_{\mu} J}(\mu) := \left( \Delta t \sum_{k=1}^{K} \| f_{\mu_k} (\cdot; \mu) \|_{X'}^2 \right)^{\frac{1}{2}} \frac{\Delta_{N,1}^{\text{du}}(\mu)}{\sqrt{\alpha_{LB}(\mu)}}
+ \frac{\gamma_{\text{UB}}(\mu)}{\alpha_{LB}(\mu)} \Delta_{N,K}^{\text{pr}}(\mu) \Delta_{N,1}^{\text{du}}(\mu) + \frac{\Delta t}{\alpha_{LB}(\mu)} \Delta_{N,K}^{\text{pr}}(\mu) \left( \Delta t \sum_{k=1}^{K} \| p_N^k (\mu) \|_{X}^2 \right)^{\frac{1}{2}}
+ \frac{\gamma_{\text{UB}}(\mu)}{\alpha_{LB}(\mu)} \Delta_{N,1}^{\text{du}}(\mu) \left( \Delta t \sum_{k=1}^{K} \| u_N^k (\mu) \|_{X}^2 \right)^{\frac{1}{2}}. \tag{3.46}
\]

and \( \Delta_{N,K}^{\text{pr}}(\mu) \) and \( \Delta_{N,1}^{\text{du}}(\mu) \) are defined in Lemma 2.

**Proof.** Similar to the elliptic case, we define \( e_{\nabla_{\mu} J}(\mu) \) to be the error in the derivative of the cost with respect to \( \mu_i \), the ith element of the parameter vector \( \mu \). It then
follows from eq. (2.16) and eq. (3.20) that

\[ e^{\nabla_{\mu_i} J}(\mu) = \Delta t \sum_{k=1}^{K} \left\{ f_{\mu_i}(e^k_{du}(\mu); \mu) - a_{\mu_i}(u^k(\mu), p^k(\mu); \mu) - a_{\mu_i}(u_N^k(\mu), p_N^k(\mu); \mu) \right\}. \]

(3.47)

Follow the same steps as in the proof of theorem 2 we arrive at

\[ e^{\nabla_{\mu_i} J}(\mu) = \Delta t \sum_{k=1}^{K} \left\{ \| f_{\mu_i}(\cdot; \mu) \|_X \| e^k_{du}(\mu) \|_X + \gamma^{UB}_{a_{\mu_i}}(\mu) \| e^k_{pr}(\mu) \|_X \right\} \]

(3.48)

Invoking the Cauchy-Schwarz inequality and lemma 2, we obtain eq. (3.46).

\[ \square \]

3.3 Computational procedure

Like other model reduction methods, the RB method is traditionally divided into a computationally expensive offline phase and a computationally efficient online phase. During the offline phase, the reduced basis is built incrementally using solution snapshots taken from a training set, and the high-dimensional FE operators are projected onto the reduced space. In the online phase, the low-dimensional projected operators are used to efficiently obtain solution approximations and error bounds at any parameter within the domain.

Section 3.3.1 presents the offline-online decomposition for the reduced basis approximation in the elliptic case. Section 3.3.2 presents the offline-online decomposition for the new a posteriori error estimation in the elliptic case. Section 3.3.3 covers the parabolic case, focusing on the ways in which it differs from the elliptic case. Section 3.3.4 summarizes the main ingredients and costs.
3.3.1 Offline-online decomposition for reduced basis approximation

In the offline phase, a parameter training set \( \mathcal{D}_{\text{train}} \subset \mathcal{D} \) is defined, and the reduced basis is built incrementally using snapshots taken from PDE solutions at parameters in \( \mathcal{D}_{\text{train}} \). The parameters at which snapshots are taken are chosen sequentially using a greedy algorithm. At the \( n \)th step, the algorithm finds the parameters \( \mu_{n}^{pr}, \mu_{n}^{du} \in \mathcal{D}_{\text{train}} \) at which the \textit{a posteriori} error bound is highest. To update the reduced basis in the elliptic case, the snapshots \( u(\mu_{n}^{pr}) \) and \( p(\mu_{n}^{du}) \) are computed, orthonormalized, and added directly to the basis. The greedy procedure terminates when the maximum \textit{a posteriori} error bound across the training set is lower than tolerances determined by the user. We summarize this process in Algorithm 1 and discuss further computational details below.

Algorithm 1 Offline primal-dual greedy procedure

1: Define \( \mathcal{D}_{\text{train}} = \{ \mu^{(1)}, \ldots, \mu^{(M)} \} \), where \( \mu^{(i)} \in \mathcal{D} \), and let \( \tau_{\text{pr}}, \tau_{\text{du}} > 0 \)
2: Initialize primal and dual reduced bases at \( \mu^{(1)} \in \mathcal{D}_{\text{train}} \)
3: while \( \max_{\mu \in \mathcal{D}_{\text{train}}} \Delta_{N}^{\text{pr}}(\mu) > \tau_{\text{pr}} \) or \( \max_{\mu \in \mathcal{D}_{\text{train}}} \Delta_{N}^{\text{du}}(\mu) > \tau_{\text{du}} \) do
4: \( \mu_{n}^{pr} \leftarrow \arg \max_{\mu \in \mathcal{D}_{\text{train}}} \Delta_{N}^{\text{pr}}(\mu) \)
5: \( \mu_{n}^{du} \leftarrow \arg \max_{\mu \in \mathcal{D}_{\text{train}}} \Delta_{N}^{\text{du}}(\mu) \)
6: Update the primal reduced basis at \( \mu_{n}^{pr} \)
7: Update the dual reduced basis at \( \mu_{n}^{pr} \)
8: end while

To compute snapshots during the offline phase, the finite element solutions \( u(\mu_{n}^{pr}) \) and \( p(\mu_{n}^{du}) \) are computed for each \( n \in \{1, \ldots, N\} \). For the elliptic case, the following algebraic analogues of Equation (2.9b) and Equation (2.11) are solved:

\[
\mathbf{A}_{N}(\mu)\mathbf{u}_{N}(\mu) = \mathbf{F}_{N}(\mu) \tag{3.49a}
\]
\[
\mathbf{A}_{N}^{T}(\mu)\mathbf{p}_{N}(\mu) = 2\mathbf{D}_{N}\mathbf{u}_{N}(\mu) + \mathbf{L}_{N}, \tag{3.49b}
\]

where we use the subscript \( \cdot_{N} \) to emphasize that we are working in the \( \mathcal{N} \)-dimensional FE space, i.e., \( \mathbf{A}_{N}(\mu), \mathbf{D}_{N} \in \mathbb{R}^{\mathcal{N} \times \mathcal{N}} \) and \( \mathbf{F}_{N}(\mu), \mathbf{L}_{N} \in \mathbb{R}^{\mathcal{N}} \). We note that our affine
assumptions (2.6) allow \( A_N(\mu) \) and \( F_N(\mu) \) to be given by

\[
A_N(\mu) = \sum_{q=1}^{Q_e} \Theta^q(\mu) A^q_N, \quad F_N(\mu) = \sum_{q=1}^{Q_f} \Theta^q(\mu) F^q_N,
\]

and note that \( A^T_N(\mu) = A_N(\mu) \) due to symmetry. Finally, we introduce the parameter-independent matrix \( X_N \in \mathbb{R}^{N \times N} \) associated with our inner product.

At \( n = 1 \) we define the basis vectors by normalizing the primal and dual solutions:

\[
\zeta_1 = \frac{u_N(\mu_1^{pr})}{\|u_N(\mu_1^{pr})\|_X}, \quad \psi_1 = \frac{p_N(\mu_1^{du})}{\|p_N(\mu_1^{du})\|_X}.
\]

and initialize the primal and dual basis matrices \( Z_N^{pr} = \zeta_1 \) and \( Z_N^{du} = \psi_1 \). Basis vectors for subsequent \( n \) are generated by orthonormalizing the primal and dual solutions with respect to the existing basis vectors:

\[
\zeta_n = \frac{u_N(\mu_n^{pr}) - \sum_{m=1}^{n-1} (u_N(\mu_n^{pr})^T X_N \zeta_m) \zeta_m}{\|u_N(\mu_n^{pr}) - \sum_{m=1}^{n-1} (u_N(\mu_n^{pr})^T X_N \zeta_m) \zeta_m\|_X}, \quad \psi_n = \frac{p_N(\mu_n^{du}) - \sum_{m=1}^{n-1} (p_N(\mu_n^{du})^T X_N \psi_m) \psi_m}{\|p_N(\mu_n^{du}) - \sum_{m=1}^{n-1} (p_N(\mu_n^{du})^T X_N \psi_m) \psi_m\|_X}.
\]

Basis vectors are then concatenated to the existing bases, yielding \( Z_N^{pr}, Z_N^{du} \in \mathbb{R}^{N \times N} \) given by

\[
Z_N = \begin{bmatrix}
\zeta_1 & \cdots & \zeta_N \\
\vdots & \ddots & \vdots \\
\psi_1 & \cdots & \psi_N
\end{bmatrix}, \quad Y_N = \begin{bmatrix}
\zeta_1 & \cdots & \zeta_N \\
\psi_1 & \cdots & \psi_N
\end{bmatrix}.
\]

To solve Equations (3.1) and (3.2) for the reduced basis primal and dual solutions, we solve the systems

\[
A_N^{pr}(\mu) u_N(\mu) = F_N^{pr}(\mu),
\]

\[
A_N^{du}(\mu) p_N(\mu) = 2D_N^{pd} u_N(\mu) + L_N^{du},
\]

noting that \( A_N^{pr}(\mu) \), \( A_N^{du}(\mu) \), and \( F_N^{pr}(\mu) \) inherit the affine assumptions on \( A_N(\mu) \) and
\[ F_N(\mu), \text{i.e.,} \]
\[ A_{N}^{pr}(\mu) = \sum_{q=1}^{Q_a} \Theta^p_a(\mu) A_{N}^{pr,q} \]  \hfill (3.57)
\[ A_{N}^{du}(\mu) = \sum_{q=1}^{Q_a} \Theta^d_a(\mu) A_{N}^{du,q} \]  \hfill (3.58)
\[ F_{N}^{pr}(\mu) = \sum_{q=1}^{Q_f} \Theta^p_f(\mu) F_{N}^{pr,q} \]  \hfill (3.59)

where \( A_{N}^{pr,q}, A_{N}^{du,q} \in \mathbb{R}^{N \times N} \) and \( F_{N}^{q} \in \mathbb{R}^{N} \) are parameter-independent quantities given by

\[ A_{N,i,j}^{pr,q} = a^q(\zeta_i, \zeta_j; \mu), \quad 1 \leq i, j \leq N, 1 \leq q \leq Q_a \]
\[ A_{N,i,j}^{du,q} = a^q(\psi_i, \psi_j; \mu), \quad 1 \leq i, j \leq N, 1 \leq q \leq Q_a \]
\[ F_{N,i}^{q} = f^q(\zeta_i; \mu), \quad 1 \leq i \leq N, 1 \leq q \leq Q_f, \]

or, equivalently, expressed as matrices,

\[ A_{N,i,j}^{pr,q} = Z^T_N A_{N}^q Z_N, \quad A_{N,i,j}^{du,q} = Y^T_N A_{N}^q Y_N, \quad F_{N}^{pr,q} = Z^T_N F_{N}^q. \]

Similarly, \( D_{N}^{qd} \in \mathbb{R}^{N \times N} \) and \( L_{N}^{du} \in \mathbb{R}^{N} \) are given by

\[ D_{N,i,j}^{qd} = d(\zeta_i, \psi_j) \quad 1 \leq i, j \leq N, \]
\[ L_{N,i}^{du} = \ell(\zeta_i), \quad 1 \leq i \leq N, \]

where the superscript \( .^{qd} \) indicates that this reduced basis quantity is calculated in terms of both the dual and the primal basis. Equivalently,

\[ D_{N}^{qd} = Y^T_N D_N Z_N, \quad L_{N}^{du} = Y^T_N L_N. \]

To arrive at an offline-online decomposition, we note that the quantities \( A_{N}^{pr,q}, A_{N}^{du,q}, D_{N}^{qd}, F_{N}^{pr,q}, \) and \( L_{N}^{du} \), which require \( N \)-dimensional matrix manipulations to
compute, are parameter-independent, and so we may pre-compute these quantities in the offline phase. During the online phase, we need only compute the affine sums (3.57-3.59), which are $N$-dimensional calculations, in order to solve the reduced basis equations, (3.55) and (3.56). Thus, the reduced basis solutions may be calculated efficiently using only $N$-dimensional computations during the online phase.

3.3.2 Offline-online decomposition for a posteriori error estimation

We may develop a similar offline-online computational procedure for the a posteriori error estimation. To do so, we first discuss computation of the coercivity and continuity constants, and then present the standard procedure for computation of the primal bound as well as the procedures for computation of the new dual, cost, and cost gradient bounds developed in Section 3.2.2.

Preliminaries

As stated in Section 2.1, for a posteriori error estimation we require access to $\alpha_{LB}(\mu)$, a lower bound on $\alpha(\mu)$, and to $\gamma^UB(\mu)$, an upper bound on $\gamma_{a\mu}(\mu)$. If the bilinear form $a(w, v; \mu)$ is symmetric and parametrically coercive, i.e. the $\Theta^q(\mu) > 0, \forall \mu \in \mathcal{D}, \ 1 \leq \alpha \leq Q_a$ and $a^q(v, v) \geq 0$, $\forall \mu \in \mathcal{X}, 1 \leq q \leq Q_a$, we may obtain the coercivity lower bound for example via the "min-theta" approach [25], i.e. if we specify the inner product $(\cdot, \cdot)_{\mathcal{X}} = a(\cdot, \cdot; \bar{\mu})$ for some reference parameter $\bar{\mu}$, and can then choose

$$\alpha_{LB}(\mu) = \min_{\bar{\mu} \in \mathcal{Q}_a} \frac{\Theta^q(\mu)}{\Theta^q(\bar{\mu})}$$

(3.60)

This is also the approach used in the numerical results in Chapter 5. A similar approach can be used to compute $\gamma^UB_{a\mu}(\mu)$ if the derivative bilinear forms $a_{a\mu}(\cdot, \cdot; \mu)$ are parametrically coercive. However, in the more general setting the successive constraint method may be used [16]. We also require access to the continuity constant $\gamma_d$, which may be obtained via a generalized eigenvalue problem. To this end, given $v \in \mathcal{X}$, we first define the supremizer $\rho_v = \arg \sup_{w \in \mathcal{X}} \frac{d(v, w)}{\|w\|_{\mathcal{X}}}$, note that $(\rho_v, w)_{\mathcal{X}} = \ldots$
\(d(v, w), \forall v \in X\), from the Riesz representation theorem, and thus

\[
\sup_{w \in X} \frac{d(v, w)}{\|w\|_X} = \|\rho_v\|_X.
\]

It follows that \(\gamma_d^2\) is the maximum eigenvalue of the generalized eigenproblem

\[
\gamma_d^2 = \sup_{v \in X} \left( \frac{\|\rho_v\|_X}{\|v\|_X} \right)^2 = \sup_{v \in X} \frac{(\rho_v, \rho_v)_X}{(v, v)_X}.
\]

Algebraically,

\[
\gamma_d^2 = \sup_{v \in X} \frac{v^T D_N^T X^{-1} D_N v}{v^T X v} \quad \text{(3.61)}
\]

Since the bilinear form \(d\) is parameter independent, we can compute \(\gamma_d\) offline a single time before the optimization.

**Computation of error bounds**

Now that we are able to compute \(\gamma_d, \alpha_{LB}, \text{ and } \gamma_{UB}^\alpha\), we may compute our \textit{a posteriori} error bounds. We begin by recalling the primal residual (Definition 1) and use our affine definitions to expand it as

\[
\begin{align*}
    r_{pr}^p(v; \mu) &= \sum_{q=1}^{Q_f} \Theta_f^q(\mu) f^q(v) - \sum_{q=1}^{Q_a} \sum_{n=1}^N \Theta_a^q(\mu) u_{N,n}(\mu) a^q(\zeta_n, v; \mu). \\

\end{align*}
\]

We then define the following parameter-dependent vector

\[
\mathcal{E}_N^p(\mu) = (\Theta_f^1(\mu), \ldots, \Theta_f^Q_f(\mu), \Theta_a^1(\mu) u_{N,1}(\mu), \ldots, \Theta_a^Q_a(\mu) u_{N,1}, \ldots, \Theta_a^{Q_a}(\mu) u_{N,1}, \ldots) \quad \text{(3.63)}
\]

and the following parameter-independent quantity

\[
\begin{align*}
    h_N^p(v) &= (f^1(v), \ldots, f^{Q_f}(v), -a^1(\zeta_1, v; \mu), \ldots, -a^{Q_a}(\zeta_1, v; \mu), \ldots, -a^{Q_a}(\zeta_N, v; \mu), \ldots) \\
    &= (f^1(v), \ldots, f^{Q_f}(v), -a^1(\zeta_1, v; \mu), \ldots, -a^{Q_a}(\zeta_N, v; \mu))^T \quad \text{(3.64)}
\end{align*}
\]

\[41\]
such that, for $Q^p_N = Q_f + Q_a N$

$$r^p(v; \mu) = \sum_{n=1}^{Q_N} \mathcal{E}_{N,n}^p (\mu) h^p_{N,n}(v)$$

Thus,

$$\tilde{r}^p(\mu) = \sum_{n=1}^{Q_N} \mathcal{E}_{N,n}^p (\mu) \tilde{g}^p_{N,n}$$

where $(\tilde{g}^p_{N,n}, v)_X = h^p_{N,n}(v), \forall v \in X, 1 \leq n \leq Q^p_N$. We now introduce the parameter-independent matrices $H^p_{N,n} \in \mathbb{R}^{N \times Q^p_N}$ and $G^p_N \in \mathbb{R}^{Q^p_N \times Q^p_N}$:

$$H^p_{N,n} = h^p_{N,n}(\phi_i), \quad G^p_N = (H^p_N)^T X^{-1} H^p_N$$

where $\phi_i$ is the $i$th basis function in our FE truth approximation space, and $X$ is the truth inner product matrix. We can then compute $G^p_N$ offline and store it, so that the online computation of the norm

$$\|\tilde{r}^p(\mu)\|_X = \left( (\mathcal{E}^p_N(\mu))^T G^p_N \mathcal{E}^p_N(\mu) \right)^{\frac{1}{2}}$$

depends only on $N$ and not $N$.

The procedure for online computation of the norm $\|\tilde{r}^{du}(\mu)\|_X$ is analogous. We begin with the dual residual definition (3.6) and write it using the affine expansions as

$$r^{du}(v; \mu) = 2 \sum_{n=1}^{N} d(\zeta_n, v) u_{N,n}(\mu) + \ell(v) - \sum_{q=1}^{Q_a} \sum_{n=1}^{N} \Theta^q_a(\mu) p_{N,n}(\mu) a^q(\psi_n, v),$$

and define the parameter-dependent vector

$$\mathcal{E}^{du}_N(\mu) = (1, 2u_{N,1}(\mu), \ldots, 2u_{N,N}(\mu), \Theta^1_a(\mu) u_{N,1}(\mu), \ldots, \Theta^{Q_a}_a(\mu) u_{N,1}(\mu), \Theta^1_a(\mu) u_{N,N}(\mu), \ldots, \Theta^{Q_a}_a(\mu) u_{N,N})^T,$$
as well as the parameter-independent matrix

\[
\mathbf{h}^{du}_N(v) = (\ell(v), d(\zeta_1, v), \ldots, d(\zeta_N, v), -a^1(\psi_1, v; \mu), \ldots, -a^{Q_a}(\psi_1, v; \mu), \ldots, -a^1(\psi_N, v; \mu), \ldots, -a^{Q_a}(\psi_N, v; \mu))^T,
\]

which allows us to express the dual residual as

\[
\mathbf{r}^{du}(v; \mu) = \sum_{n=1}^{Q_{du}^N} \mathcal{E}^{du}_{N,n}(\mu) \mathbf{h}^{du}_{N,n}(v)
\]

for \(Q_{du}^N = 1 + (Q_a + 1)N\). It follows from the Riesz representation of \(\mathbf{h}^{du}_{N,n}(v)\),

\[
(\mathcal{G}^{du}_{N,n}, v)_X = \mathbf{h}^{du}_{N,n}(v), \quad \forall v \in X, 1 \leq n \leq Q_{du}^N,
\]

that

\[
\mathbf{r}^{du}(\mu) = \sum_{n=1}^{Q_{du}^N} \mathcal{E}^{du}_{N,n}(\mu) \mathcal{G}^{du}_{N,n},
\]

and thus, we may introduce the parameter-independent matrices \(\mathbf{H}^{du}_N \in \mathbb{R}^{N \times Q_{du}^N}\) and \(\mathbf{G}^{du}_N \in \mathbb{R}^{Q_{du}^N \times Q_{du}^N}\),

\[
\mathbf{H}^{du}_{N,n} = \mathbf{h}^{du}_{N,n}(\phi_1), \quad \mathbf{G}^{du}_N = (\mathbf{H}^{du}_N)^T X^{-1} \mathbf{H}^{du}_N
\]

allowing offline computation of \(\mathbf{G}^{du}_N\), such that the online computation of the norm

\[
\|\mathbf{r}^{du}(\mu)\|_X = \left( (\mathcal{E}^{du}_N(\mu))^T \mathbf{G}^{du}_N \mathcal{E}^{du}_N(\mu) \right)^{\frac{1}{2}}
\]

involves only \(N\)-dimensional quantities.

We now have efficient methods of calculating the \(X\)-norm error bounds \(\Delta^{pr}_N(\mu)\) and \(\Delta^{du}_N(\mu)\). However, to calculate the error bounds on the cost and cost gradient efficiently, we additionally require efficient methods for computing \(|\mathbf{r}^{pr}(p_N(\mu); \mu)|\).
\[ \|u_N(\mu)\|_X, \text{ and } \|p_N(\mu)\|_X. \] To do so, we note that

\[
r_{pr}(p_N(\mu); \mu) = \sum_{q=1}^{Q_f} \Theta_j^q(\mu) f^q(p_N(\mu)) - \sum_{q=1}^{Q_a} \sum_{n=1}^N \Theta_q^q(\mu) u_{N,n}(\mu) a^q(\xi_n, p_N(\mu); \mu)
\]

\[
= \sum_{q=1}^{Q_f} \sum_{n=1}^N \Theta_j^q(\mu) p_{N,n}(\mu) f^q(\psi_n) - \sum_{q=1}^{Q_a} \sum_{n=1}^N \Theta_q^q(\mu) u_{N,n}(\mu) p_{N,n}(\mu) a^q(\xi_n, \psi_n; \mu)
\]

This follows from (3.62). Equivalently, in matrix notation, we have

\[
r_{pr}(p_N(\mu); \mu) = \left( \sum_{q=1}^{Q_f} \Theta_j^q(\mu) (p_N(\mu))^T F_{N}^{du,q} - \sum_{q=1}^{Q_a} \Theta_q^q(\mu) (p_N(\mu))^T A_N^{pd,q} u_N(\mu) \right)
\]

where the \( N \)-dimensional quantities \( F_{N}^{du,q} \) and \( A_N^{pd,q} \) may be calculated offline via

\[
F_{N}^{du,q} = Y_N^T F_q
\]

and

\[
A_N^{pd,q} = Y_N^T A_N^q Z_N,
\]

allowing efficient (\( N \)-dimensional) computation of \( |r_{pr}(p_N(\mu); \mu)|. \)

For the terms \( \|u_N(\mu)\|_X \) and \( \|p_N(\mu)\|_X \), we note that

\[
\|u_N(\mu)\|_X = \sqrt{(u_N(\mu))^T X u_N(\mu)} = \sqrt{(u_N(\mu))^T Z_N^T X Z_N u_N(\mu)}
\]

and

\[
\|p_N(\mu)\|_X = \sqrt{(p_N(\mu))^T X p_N(\mu)} = \sqrt{(p_N(\mu))^T Y_N^T X Y_N p_N(\mu)}.
\]

Thus, if we precompute \( X_{N}^{du} = Z_N^T X Z_N \) and \( X_{N}^{du} = Y_N^T X Y_N \) in the offline phase, the norms \( \|u_N(\mu)\|_X \) and \( \|p_N(\mu)\|_X \) may be calculated efficiently in the online phase.

### 3.3.3 Parabolic case

In the parabolic case, the greedy procedure described in described in Section 3.3.1 is replaced with a POD-greedy procedure. During the offline phase, the algebraic
analogue of Equations (3.17) and (3.18),

\[
\frac{1}{\Delta t} M_N(u^k_N(\mu) - u^{k-1}_N(\mu)) + A_N u^k_N(\mu) = F_N,
\]

\[
\frac{1}{\Delta t} M_N(p^k_N(\mu) - p^{k+1}_N(\mu)) + A_N^T p^k_N(\mu) = 2D_N u^k_N(\mu) + L_N,
\]

are solved at the parameters $\mu^{pr}_n$ and $\mu^{du}_n$ to yield $u^k_N(\mu^{pr}_n)$ and $p^k_N(\mu^{du}_n)$, for $k \in K$. The $X$-orthogonal projections of $u^k_N(\mu^{pr}_n)$ and $p^k_N(\mu^{du}_n)$ onto the current bases are computed, and then the largest POD modes of the primal and dual projection errors are orthonormalized and added to the basis as in eqs. (3.51)-(3.54). The finite element operators $A_N$, $F_N$, $D_N$, and $L_N$ are then projected onto $Z_N$ and $Y_N$ to obtain the reduced basis operators $A_N^{pr,q}$, $A_N^{du,q}$, $D_N^{pd}$, $F_N^{pr,q}$, and $L_N^{du}$ as in the elliptic case. Additionally, we project the finite element operator $M_N$ onto $Z_N$ and $Y_N$ to obtain:

\[
M_N^{pr} = Z_N^T M_N Z_N
\]

\[
M_N^{du} = Y_N^T M_N Y_N.
\]

We may then solve the algebraic systems

\[
\frac{1}{\Delta t} M_N^{pr}(u^k_N(\mu) - u^{k-1}_N(\mu)) + A_N^{pr} u^k_N(\mu) = F_N^{pr},
\]

\[
\frac{1}{\Delta t} M_N^{du}(p^k_N(\mu) - p^{k+1}_N(\mu)) + A_N^{du} p^k_N(\mu) = 2D_N^{pd} u^k_N(\mu) + L_N^{du},
\]

for the reduced basis solutions $u^k_N(\mu)$ and $p^k_N(\mu)$ at any parameter $\mu$ efficiently during the online phase.

Similar to the elliptic case, we require efficient methods of computing $\|r^{k}_{pr}(\cdot; \mu)\|^2_X$, $\|r^{k}_{du}(\cdot; \mu)\|^2_X$, $\|r^{k}_{pN}(\mu; \mu)\|$, $\|u^k_N(\mu)\|_X$, and $\|p^k_N(\mu)\|_X$. Computation of these quantities is exactly analogous to the elliptic case, with added terms due to the bilinear form $m$, and is not detailed here.
3.3.4 Summary of costs

The costs of calculating the FE snapshots during the offline phase are thus $2N \mathcal{N}$-dimensional $A(\mu)$-solves (one primal and one dual solve for each $N$) for the elliptic case, and $2NK \mathcal{N}$-dimensional $A(\mu)$-solves in the parabolic case (the cost of time integration without LU-factorization for both the dual and the primal for each $N$). Here, $A(\mu)$ is the FE matrix corresponding to the bilinear form $a$.

Additionally, in order to facilitate efficient online error estimation, the offline phase requires $(Q_a + Q_f) \mathcal{N}$-dimensional solves of the $X$-inner product matrix (denoted $X$) per vector added to the basis. Since the matrix $X$ is parameter-independent, we may precompute its (sparse) LU-factorization once at the start of the optimization, allowing the necessary $X$-solves to be efficiently executed offline.
Chapter 4

Trust region reduced basis algorithm

This chapter introduces the combined trust region reduced basis approach to PDE-constrained optimization. Section 4.1 discusses general trust region methods and introduces our framework. Section 4.2 presents the convergence theory for our method, using the new cost and cost gradient bounds developed in Chapter 3, and Section 4.3 details the algorithm.

4.1 Trust region framework

The canonical trust region optimization framework solves a set of successive optimization subproblems, defined as

$$\min_s M_k^k(\mu^k + s) \quad \text{s.t.} \|s\| \leq \delta^k,$$

where $\mu^k$ is the current optimization iterate, $M_k^k(\mu)$ is the model function used to approximate the true objective function $J(\mu)$, $\delta^k$ is the trust region radius, and we solve for $s$, the optimal step within the defined trust region; we refer to the book [9] for an extensive resource on trust region methods. The model function $M_k^k(\mu)$ changes at each trust region iteration and is often a local quadratic Taylor expansion. Other surrogates, however, have also been considered in the literature [1, 2, 17, 19, 30].

To determine if the step $s$ should be accepted, the ratio $\rho^k = \frac{M(\mu^k) - M(\mu^{k+1})}{J(\mu^k) - J(\mu^{k+1})}$,
a measure of how well the model predicts decrease in the true cost, is computed. The value of $\rho^k$ is used to determine not only whether or not the optimization step is accepted, but also whether and how to change trust region radius for the next optimization subproblem. One criticism of this approach in the POD or general surrogate model context is that the computation of $\rho^k$ requires evaluating the true objective function $J(\mu)$, which may be computationally expensive [2, 30].

Standard trust region convergence theory requires (i) that the model function $m^k$ satisfy the first-order condition, i.e. the model function must match the true objective and gradient at the current iterate exactly, and (ii) that each iterate of the optimization meet a sufficient decrease condition. It has been shown, however, that trust region optimizations converge even if inexact model and gradient information is used [7, 13, 26].

In this work, the reduced basis cost $J^k(\mu)$ serves as the model function $M^k(\mu)$. The a posteriori error bounds developed in Chapter 3 are used (i) to minimize the number of true objective evaluations required, and (ii) together with a recent result from Yue and Meerbergen [30], to guarantee convergence of the approach to the optimum of the high-fidelity model.

We stress here that our approach breaks from the traditional RB offline/online strategy here, i.e. we generate the RB approximation on the fly during the optimization: we use the online evaluation to efficiently solve the trust region subproblems and update the reduced basis along the optimization trajectory only if the a posteriori error bounds indicate a need to do so. The offline and online stages thus intertwine and each reduced basis update requires an FE snapshot computation and update of the error bound computation as discussed in Chapter 3.

### 4.2 Convergence

In [30], Yue and Meerbergen relax the stringent first-order accuracy requirements to consider the general setting of an unconstrained trust region optimization algorithm using surrogate models with the following properties:
1. a bound on the error in the model function exists over the entire parameter space,

2. at any point within the parameter domain, we may reduce the approximation error to within any given tolerance $\epsilon > 0$, and

3. the model function must be smooth with finite gradient everywhere.

Given the above conditions, [30] replaces the first-order condition with the following relaxed first-order condition (adapted to our notation from Chapters 2 and 3):

\[
|J_N^k(\mu^k) - J(\mu^k)| \leq \Delta_N^{J,k}(\mu^k) \quad \text{and} \quad \left\| \nabla_{\mu} J_N^k(\mu^k) - \nabla_{\mu} J(\mu^k) \right\| \leq \Delta_N^{\nabla J,k}(\mu^k) \tag{4.1a}
\]

\[
\frac{\Delta_N^{J,k}(\mu^k)}{J_N^k(\mu^k)} \leq \tau_J \quad \text{and} \quad \frac{\Delta_N^{\nabla J,k}(\mu^k)}{\left\| \nabla_{\mu} J_N^k(\mu^k) \right\|} \leq \tau_{\nabla J} \tag{4.1b}
\]

for any given $\tau_J > 0$ and $\tau_{\nabla J} > 0$. There are two parts to this condition: Equation (4.1a) requires that error bounds exist for both the cost function and its gradient, while Equation (4.1b) requires that the reduced basis model be able to meet arbitrarily small tolerances $\tau_J$ and $\tau_{\nabla J}$. The sufficient decrease condition is similarly replaced, with an "error-aware sufficient decrease condition":

\[
J_N^{k+1}(\mu^{k+1}) \leq J_N^k(\mu_{AGC}) \tag{4.2}
\]

where $\mu_{AGC}$ is known as the "approximate generalized Cauchy point", a point that achieves sufficient decrease in the reduced basis model in a descent direction.

To ensure that all optimization iterates satisfy the error-aware sufficient decrease condition, Yue and Meerbergen present a procedure designed to reject steps which violate this condition [30]. We summarize the procedure in [30] using our notation here and begin by noting that a sufficient condition for (4.2) is

\[
J_N^k(\mu^{k+1}) + \Delta_N^{J,k}(\mu^{k+1}) + \Delta_N^{J,k+1}(\mu^{k+1}) \leq J_N^k(\mu_{AGC}).
\]
However, we do not have access to $\Delta_{N}^{jk+1}(\mu^{k+1})$. Instead, it is sufficient to check

\[ J_{N}^{k}(\mu^{k+1}) + \Delta_{N}^{j,k}(\mu^{k+1}) < J_{N}^{k}(\mu_{AGC}^{k}), \]  

(4.3)

because we may update the RB model with basis functions taken at $\mu^{k+1}$ before the next subproblem solve to ensure that $\Delta_{N}^{jk+1}(\mu^{k+1}) = 0$, thus satisfying the sufficient condition. We can check this cheaply, and if it holds we may accept the iterate $\mu^{k+1}$, updating the reduced basis model at $\mu^{k+1}$ as necessary.

Otherwise, we note that a necessary condition for (4.2) is

\[ J_{N}^{k}(\mu^{k+1}) - \Delta_{N}^{j,k}(\mu^{k+1}) - \Delta_{N}^{j,k+1}(\mu^{k+1}) \leq J_{N}^{k}(\mu_{AGC}^{k}), \]  

(4.4)

so we check

\[ J_{N}^{k}(\mu^{k+1}) - \Delta_{N}^{j,k}(\mu^{k+1}) \leq J_{N}^{k}(\mu_{AGC}^{k}). \]  

(4.5)

If this condition fails, satisfying (4.4) may require a large error bound in the next model, leading to inaccurate approximations, so we reject the iterate $\mu^{k+1}$, shrink the trust region (set $\epsilon_{L} = \kappa_{tr} \epsilon_{L}$ for some $\kappa_{tr} \in (0, 1)$), and re-solve the optimization sub-problem. Otherwise, if (4.5) holds, we update the model at $\mu^{k+1}$ and check (4.2). If it holds, then we accept $\mu^{k+1}$. Otherwise, we reject $\mu^{k+1}$, shrink the trust region, and re-solve the optimization subproblem.

If the relaxed first-order condition is satisfied, and all iterates satisfy the error-aware sufficient decrease condition, Yue and Meerbergen show convergence of the trust region algorithm to the optimum of the high-fidelity model under mild assumptions [30]: besides constraints on the parameters for the trust region algorithm summarized in Table 5.1, which are easily satisfied, we also require lower boundedness of the cost functional and Lipschitz continuity of cost gradient and the constraints. However, given our assumptions on the problem formulation in Chapter 2, these requirements are satisfied in our setting.
4.3 Algorithm

The optimization subproblem for the trust region reduced basis algorithm is defined as follows:

\[
\min_{\mu^{k+1}} J_N^k (\mu^{k+1}) \quad \text{s.t.} \quad \left| \frac{\Delta_N^{j,k}(\mu^{k+1})}{J_N^k(\mu^{k+1})} \right| \leq \epsilon_L
\]

(4.6)

where \( \epsilon_L \) is the maximum allowable relative error in the cost. We note that the error bound on the cost functional, \( \Delta_N^{j,k}(\mu) \), is used to implicitly define the trust region; if the subproblem solver steps outside of this region, we use backtracking to return to a region where \( \Delta_N^{j,k}(\mu) \) is sufficiently low.

For each subproblem solve, we have two possible termination criteria: either (a) the line search method locates a stationary point within the trust region, or (b) the line search gets close to the boundary of the current trust region, i.e.,

\[(a) \quad \| \nabla J_N^k(\mu) \| \leq \tau_{sub} \quad \text{or} \quad \quad (b) \quad \beta \epsilon_L \leq \frac{\Delta_N^{j,k}(\mu)}{J_N^k(\mu)} \leq \epsilon_L \]

(4.7)

for some small \( \tau_{sub} \geq 0 \) and for some \( \beta \in (0, 1) \), generally close to 1. The latter criterion prevents the algorithm from expending too much effort optimizing close to the trust region boundary where the model becomes inaccurate. Overall convergence is reached when the norm of the true gradient is less than a tolerance \( \tau \geq \tau_{sub} \), i.e.

\[\| \nabla J(\mu^k) \| \leq \| \nabla J_N^k(\mu^k) \| + \Delta_N^{\nabla,\mu,j,k}(\mu^k) \leq \tau.\]

The reduced model employed is an iteratively-built reduced basis model that is updated only when the subproblem optimization terminates on condition (4.7b), indicating that our RB model is not sufficiently accurate. In the elliptic case, updating the RB model entails adding \( u(\mu^k) \) and \( \psi(\mu^k) \) to the primal and dual bases. In doing so, we automatically satisfy Equation (4.1b), since the reduced basis is able to exactly represent the FE solution \( \mu^k \). In the parabolic case, we may add singular modes from the primal and dual solutions at the current iterate until Equation (4.1b) is satisfied.
The algorithm steps are summarized in Algorithm 2.

**Algorithm 2 Trust region reduced basis optimization**

1: Initialize. Let $k = 0$, and choose $\tau \geq \tau_{sub} \geq 0$, $\tau_{\nabla J} \in (0, 1)$, and $\beta \in (0, 1)$. Additionally, choose $\mu^0$, $\epsilon_L$, and $\kappa_{tr} < 1$, a trust region decrease factor. Initialize the initial reduced basis model at $\mu^0$.

2: Solve the optimization subproblem eq. (4.6) with termination criteria eq. (4.7).

3: if the sufficient condition eq. (4.3) holds then
   4: Accept and update the reduced model at $\mu^{k+1}$ and go to Line 15.

5: else if the necessary condition eq. (4.5) fails then
   6: Reject $\mu^{k+1}$, set $\epsilon_L = \kappa_{tr}\epsilon_L$ and return to Line 2.

7: else
   8: Update the model at $\mu^{k+1}$.
   9: if the EASDC eq. (4.2) holds then
     10: Accept $\mu^{k+1}$ and go to Line 15.
   11: else
     12: Reject $\mu^{k+1}$, set $\epsilon_L = \kappa_{tr}\epsilon_L$ and return to Line 2.
   13: end if
14: end if
15: If $\|\nabla J_N^{k+1}(\mu^{k+1})\| + \Delta_N^{\nabla J_N^{k+1}(\mu^{k+1})} \leq \tau$, return $\mu^{k+1}$ and stop. Otherwise, go to Line 2.
Chapter 5

Numerical tests

In this chapter, we present numerical results for a thermal fin model problem. Section 5.1 introduces the problem formulation. Section 5.2 presents results regarding the quality of a traditional reduced basis approximation for the thermal fin model problem. Section 5.3 compares performance of the three optimization approaches:

1. an FE-only approach, consisting of an interior point optimizer [3, 4, 29] as implemented in MATLAB's fmincon routine, using the high-dimensional FE model for its function and gradient evaluations,

2. a traditional RB approach, consisting of an offline phase, in which a global reduced basis is built, and an online phase, in which MATLAB's fmincon interior point implementation is used to solve the optimization using reduced basis function and gradient evaluations, and

3. the trust region reduced basis algorithm presented in Section 4.3, employing the BFGS quasi-Newton method to solve each trust region subproblem using only reduced evaluations, and solving the full model as needed to progressively build the reduced basis along the optimization trajectory,

for a two- and six-parameter optimization. The algorithm parameters used for the optimization tests and construction of the reduced bases for the traditional offline/online approach are shown in Table 5.1.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value for numerical tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>“close” to TR boundary threshold</td>
<td>$\beta$</td>
<td>0.95</td>
</tr>
<tr>
<td>trust region boundary</td>
<td>$\varepsilon_L$</td>
<td>0.1</td>
</tr>
<tr>
<td>RB gradient error tolerance</td>
<td>$\tau_{\nabla J}$</td>
<td>0.1</td>
</tr>
<tr>
<td>subproblem convergence tolerance</td>
<td>$\tau_{sub}$</td>
<td>1e-8</td>
</tr>
<tr>
<td>overall convergence tolerance</td>
<td>$\tau$</td>
<td>5e-4</td>
</tr>
</tbody>
</table>

Table 5.1: Trust region reduced basis algorithm parameters used in numerical tests

![Figure 5-1: Thermal fin geometry](image)

5.1 Thermal fin model problem

We consider a two-dimensional thermal fin with a fixed geometry (Fig. 5-1) consisting of a central post and four horizontal subfins, with interior $\Omega$ and boundary $\Gamma$ [20]. The fin conducts heat away from a uniform heat flux source at the root of the fin, $\Gamma_{\text{root}}$, through the post and subfins to the surrounding air. The fin is characterized by a six-dimensional parameter vector $\mu = (k_0, k_1, k_2, k_3, k_4, Bi)^T$ containing the heat conductivities, $k_i \in [0.1, 10]$, of the subfins and the central post and the Biot number, $Bi \in [0.01, 1]$, a nondimensional heat transfer coefficient relating the convective heat transfer coefficient to the conductivity of the fin. We will consider a two-parameter and a six-parameter optimization. In the two-parameter optimization, we fix $k_0 = 1$ and constrain the subfin conductivities to vary together (i.e., $k_1 = k_2 = k_3 = k_4$). In the six-parameter optimization, all six components of $\mu$ may vary independently.

5.1.1 Elliptic model problem

The temperature distribution within the fin, $u(\mu)$, is governed by the steady heat equation with a unit Neumann flux boundary condition at the root of the fin to model a heat source. Robin boundary conditions on all other external boundaries
model convective heat losses, and we enforce continuity of both $u$ and its gradient at interfaces between the fin post and subfins. The output of interest is the average temperature of the fin root, $T_{\text{root}}(\mu) = \mathcal{L}(u(\mu)) = \int_{\Gamma_{\text{root}}} u(\mu)$. For the high-fidelity model we consider a piecewise linear FE approximation space $X$ on a quasi-uniform unstructured mesh of dimension $\dim(X) = N = 17899$.

For our optimization, we generate synthetic experimental measurements $\hat{T}_{\text{root}}$ by considering a thermal fin whose parameters are fixed but unknown. We then aim to infer the unknown parameters by minimizing the output least-squares formulation

$$s(\mu) = \frac{1}{2}(T_{\text{root}}(\mu) - \hat{T}_{\text{root}})^2 = \frac{1}{2}\|\mathcal{L}(u(\mu)) - \hat{T}_{\text{root}}\|_{\mathbb{R}}^2. \quad (5.1)$$

To obtain a cost function of the form presented in Section 2.2, we define $d(u, v) \equiv \frac{1}{2}(\mathcal{L}u, \mathcal{L}v)_{\mathbb{R}}$ and $\ell(v) \equiv -\left(\mathcal{L}v, \hat{T}_{\text{root}}\right)_{\mathbb{R}}$, drop the constant term $\frac{1}{2}(\hat{T}_{\text{root}}, \hat{T}_{\text{root}})_{\mathbb{R}}$, and introduce the regularization $\mathcal{R}(\mu) = \|\frac{\mu - \hat{\mu}}{\hat{\mu}}\|_{\mathbb{R}}^2$, where $\hat{\mu} \in \mathcal{D}$.

### 5.1.2 Parabolic model problem

We now consider the time-varying temperature distribution within the fin in the time interval $I = [0, 10]$ governed by the time-dependent heat equation with a sinusoidal control input $y(t) = \cos(t)$ at the root of the fin. As in the elliptic problem, we enforce Robin boundary conditions at all other external boundaries and continuity of temperature and heat flux at all internal interfaces. In the parabolic problem, our output of interest is the average temperature of the entire fin at the current timestep, $T_{\text{avg}}^k(\mu) = \mathcal{L}(u^k(\mu)) = \int_{\Omega} u^k(\mu)$. Again, we generate artificial output data $\hat{T}_{\text{avg}}^k$ for all $k \in K$ by considering a fin whose parameters are fixed but unknown. Thus, our output least-squares formulation is given by

$$s(\mu) = \Delta t \sum_{k=1}^{K} \frac{1}{2}(T_{\text{avg}}^k(\mu) - \hat{T}_{\text{avg}}^k)^2 = \Delta t \sum_{k=1}^{K} \frac{1}{2}\|\mathcal{L}(u^k(\mu)) - \hat{T}_{\text{avg}}^k\|_{\mathbb{R}}^2. \quad (5.2)$$

Analogous to the elliptic case, we may obtain a least-squares cost functional of the form presented in Section 2.3 by defining $d(u, v) \equiv \frac{1}{2}(\mathcal{L}u, \mathcal{L}v)_{\mathbb{R}}$ and $\ell(v) \equiv$
dropping the constant term $\frac{1}{2}(\hat{T}^k_{\text{avg}}, \hat{T}_{\text{avg}})_{\mathbb{R}}$, and introducing the regularization $\mathcal{R}(\mu) = ||\mu - \hat{\mu}||^2_{\mathbb{R}}$, where $\hat{\mu} \in \mathcal{D}$.

5.1.3 Problem data

In Section 5.3, we compare the performance of our trust region algorithm to that of the FE-only and RB-only fmincon interior point approaches for the 2D and 6D elliptic and parabolic optimizations. Optimization trials in each case are run on ten different least-squares cost functionals, corresponding to ten randomly selected values for $\mu^*$ within the parameter domain. For each randomly selected $\mu^*$-value, we obtain $\hat{T}_{\text{root}}$ or $\hat{T}^k_{\text{avg}}$ from the high-fidelity FE model. The value $\mu^*$ is then used as the regularization function parameter $\hat{\mu}$. Table 5.2 specifies the data used to generate numerical results in subsequent sections.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Elliptic</th>
<th>Parabolic</th>
</tr>
</thead>
<tbody>
<tr>
<td>FE dimension</td>
<td>$N$</td>
<td>17899</td>
<td></td>
</tr>
<tr>
<td>$d$-continuity constant</td>
<td>$\gamma_d$</td>
<td>0.7999</td>
<td>9.6970</td>
</tr>
<tr>
<td>Number of time steps</td>
<td>$K$</td>
<td>–</td>
<td>100</td>
</tr>
<tr>
<td>Regularization scaling factor</td>
<td>$\lambda$</td>
<td>1</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Table 5.2: Problem data used for generation of numerical results. The $X$-inner product was defined as $(\cdot, \cdot)_X = a(\cdot, \cdot; \bar{\mu})$, for $\bar{\mu} = (1, 1, 1, 1, 1, 0.1)^T$.

5.2 Global reduced-basis approximation quality

In order to compare the performance of the proposed trust region reduced basis approach to the performance of a traditional RB approach, we generate a reduced basis offline. We introduce a training set $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ of size $n_{\text{train}}$ and an initial parameter $\mu^{(1)}$, and employ a cost-based greedy algorithm to build the reduced basis based on the $a$ posteriori error bounds on the cost and cost gradient. The exact procedure employed is given in Algorithm 3, in which the tolerance values $\tau$ and $\tau_{\gamma J}$ are those required for the optimization (see Table 5.1). This ensures that error everywhere on the training grid is low enough to meet the convergence tolerances.
However, we note that this does not guarantee low error over the entire parameter domain.

Algorithm 3 Offline cost-based greedy procedure

1: Choose $D_{\text{train}} \subset D$, $\tau > 0$, and $\tau_{\text{TVJ}} > 0$
2: Initialize primal and dual reduced bases at $\mu_1^{(1)} \in D_{\text{train}}$
3: while $\max_{\mu \in D_{\text{train}}} \frac{\Delta^J_{\text{rel}}(\mu)}{J_N(\mu)} > \tau$ or $\max_{\mu \in D_{\text{train}}} \frac{\Delta^{\nabla^J_{\mu}}(\mu)}{\|\nabla_{\mu} J_N(\mu)\|} > \tau_{\text{TVJ}}$ do
4: if $\max_{\mu \in D_{\text{train}}} \frac{\Delta^J_{\text{rel}}(\mu)}{J_N(\mu)} > \tau$ then
5: $\mu^* \leftarrow \arg \max_{\mu \in D_{\text{train}}} \frac{\Delta^J_{\text{rel}}(\mu)}{J_N(\mu)}$
6: else
7: $\mu^* \leftarrow \arg \max_{\mu \in D_{\text{train}}} \frac{\Delta^{\nabla^J_{\mu}}(\mu)}{\|\nabla_{\mu} J_N(\mu)\|}$
8: end if
9: Update the reduced-basis at $\mu^*$
10: end while

We now present the standard convergence results for both the two- and the six-parameter case. Specifically, Tables 5.3–5.6 present, as a function of $N$, the maximum relative error bounds $\Delta_{\text{rel}}^{pr}, \Delta_{\text{rel}}^{du}, \Delta_{\text{rel}}^{J}, \Delta_{\text{rel}}^{\nabla^J_{\mu}}$, as well as the average effectivities $\bar{\eta}_{\text{pr}}, \bar{\eta}_{\text{du}}, \bar{\eta}_{J},$ and $\bar{\eta}_{\text{TVJ}}$, over a randomly generated test set $\Xi \subset D$ of size $n_{\text{train}} = 100$, i.e., for the elliptic case we have $\Delta_{\text{rel}}^{pr} = \max_{\mu \in \Xi} \frac{\Delta^p_{\mu}(\mu)}{\|u(\mu)\|_X}$, $\Delta_{\text{rel}}^{du} = \max_{\mu \in \Xi} \frac{\Delta^p_{\mu}(\mu)}{\|u(\mu)\|_X}$, $\Delta_{\text{rel}}^{J} = \max_{\mu \in \Xi} \frac{\Delta^J_{\mu}(\mu)}{\|J(\mu)\|_X}$, $\Delta_{\text{rel}}^{\nabla^J_{\mu}} = \max_{\mu \in \Xi} \frac{\Delta^{\nabla_{\mu} J_{\mu}}(\mu)}{\|\nabla_{\mu} J_{\mu}(\mu)\|}$, and

$\eta_{\text{pr}}(\mu) = \frac{\Delta^p_{\mu}(\mu)}{\|u(\mu)\|_X}, \eta_{\text{du}}(\mu) = \frac{\Delta^p_{\mu}(\mu)}{\|u(\mu)\|_X}, \eta_{J}(\mu) = \frac{\Delta^J_{\mu}(\mu)}{\|J(\mu)\|_X}, \eta_{\text{TVJ}}(\mu) = \frac{\Delta^{\nabla_{\mu} J_{\mu}}(\mu)}{\|\nabla_{\mu} J_{\mu}(\mu)\|}$. The corresponding definitions for the parabolic case are similar and thus omitted.

We observe that the effectivities of the primal bounds are close to 1 for all cases considered thus indicating very sharp bounds. Dual effectivities are considerably larger, due to the propagation of the primal error to the dual problem and entering into the dual error bound formulation. The error bounds for the cost functional converge quickly, enabling us to achieve the required error tolerance for the trust region approach. Except for small $N$, the cost effectivities have a range of $O(10^{-100})$ for the elliptic case and $O(100-1000)$ for the parabolic case, which is acceptable given the fast convergence of the reduced-basis approximation. As anticipated, the bounds for the cost gradients have the highest effectivities. However, as discussed in Section 3.1.2, even fairly large relative errors in the gradient are permissible in the
trust region approach, and this result thus poses no impediment for our approach.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\Delta_{\text{rel, max}}^{pr}$</th>
<th>$\bar{\eta}_{\text{pr}}$</th>
<th>$\Delta_{\text{rel, max}}^{du}$</th>
<th>$\bar{\eta}_{\text{du}}$</th>
<th>$\Delta_{\text{rel, max}}^{J}$</th>
<th>$\bar{\eta}_{J}$</th>
<th>$\Delta_{\text{rel, max}}^{\nabla \mu J}$</th>
<th>$\bar{\eta}_{\nabla \mu J}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2.36</td>
<td>2.22</td>
<td>80.5</td>
<td>47.1</td>
<td>0.55</td>
<td>54.9</td>
<td>37.1</td>
<td>439</td>
</tr>
<tr>
<td>6</td>
<td>1.91</td>
<td>2.76</td>
<td>62.2</td>
<td>44.9</td>
<td>3.1e-2</td>
<td>53.2</td>
<td>3.38</td>
<td>605</td>
</tr>
<tr>
<td>8</td>
<td>0.64</td>
<td>2.52</td>
<td>43.4</td>
<td>44.2</td>
<td>1.2e-2</td>
<td>39.9</td>
<td>1.76</td>
<td>1047</td>
</tr>
<tr>
<td>10</td>
<td>0.41</td>
<td>3.24</td>
<td>34.1</td>
<td>48.4</td>
<td>7.5e-3</td>
<td>40.3</td>
<td>1.66</td>
<td>1882</td>
</tr>
</tbody>
</table>

Table 5.3: 2D elliptic thermal fin problem: convergence rate and effectivities of traditional reduced basis

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\Delta_{\text{rel, max}}^{pr}$</th>
<th>$\bar{\eta}_{\text{pr}}$</th>
<th>$\Delta_{\text{rel, max}}^{du}$</th>
<th>$\bar{\eta}_{du}$</th>
<th>$\Delta_{\text{rel, max}}^{J}$</th>
<th>$\bar{\eta}_{J}$</th>
<th>$\Delta_{\text{rel, max}}^{\nabla \mu J}$</th>
<th>$\bar{\eta}_{\nabla \mu J}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>5.83</td>
<td>5.39</td>
<td>8.1e2</td>
<td>85.8</td>
<td>0.84</td>
<td>250</td>
<td>323</td>
<td>2.4e3</td>
</tr>
<tr>
<td>16</td>
<td>1.70</td>
<td>6.25</td>
<td>1.2e2</td>
<td>144</td>
<td>2.0e-2</td>
<td>174</td>
<td>45.4</td>
<td>4.2e3</td>
</tr>
<tr>
<td>32</td>
<td>0.66</td>
<td>6.23</td>
<td>9.85</td>
<td>190</td>
<td>4.0e-4</td>
<td>97.9</td>
<td>0.68</td>
<td>6.6e4</td>
</tr>
<tr>
<td>48</td>
<td>0.16</td>
<td>6.88</td>
<td>2.00</td>
<td>190</td>
<td>1.8e-5</td>
<td>129</td>
<td>9.7e-2</td>
<td>4.9e5</td>
</tr>
</tbody>
</table>

Table 5.4: 6D elliptic thermal fin problem: convergence rate and effectivities of traditional reduced basis

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\Delta_{\text{rel, max}}^{pr}$</th>
<th>$\bar{\eta}_{\text{pr}}$</th>
<th>$\Delta_{\text{rel, max}}^{du}$</th>
<th>$\bar{\eta}_{du}$</th>
<th>$\Delta_{\text{rel, max}}^{J}$</th>
<th>$\bar{\eta}_{J}$</th>
<th>$\Delta_{\text{rel, max}}^{\nabla \mu J}$</th>
<th>$\bar{\eta}_{\nabla \mu J}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>0.31</td>
<td>2.05</td>
<td>2.9e2</td>
<td>5.1e2</td>
<td>0.41</td>
<td>2.1e3</td>
<td>4.2e2</td>
<td>3.5e4</td>
</tr>
<tr>
<td>30</td>
<td>3.3e-2</td>
<td>1.79</td>
<td>34.0</td>
<td>3.4e3</td>
<td>9.8e-3</td>
<td>4.3e3</td>
<td>29.8</td>
<td>2.8e4</td>
</tr>
<tr>
<td>45</td>
<td>1.2e-2</td>
<td>1.77</td>
<td>7.48</td>
<td>5.7e3</td>
<td>6.6e-4</td>
<td>7.2e2</td>
<td>7.21</td>
<td>8.4e4</td>
</tr>
<tr>
<td>60</td>
<td>5.8e-3</td>
<td>1.84</td>
<td>4.58</td>
<td>2.0e4</td>
<td>1.2e-4</td>
<td>5.3e2</td>
<td>4.40</td>
<td>4.2e5</td>
</tr>
</tbody>
</table>

Table 5.5: 2D parabolic thermal fin problem: convergence rate and effectivities of traditional reduced basis

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\Delta_{\text{rel, max}}^{pr}$</th>
<th>$\bar{\eta}_{\text{pr}}$</th>
<th>$\Delta_{\text{rel, max}}^{du}$</th>
<th>$\bar{\eta}_{du}$</th>
<th>$\Delta_{\text{rel, max}}^{J}$</th>
<th>$\bar{\eta}_{J}$</th>
<th>$\Delta_{\text{rel, max}}^{\nabla \mu J}$</th>
<th>$\bar{\eta}_{\nabla \mu J}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>4.3e-2</td>
<td>3.0</td>
<td>29</td>
<td>1.6e3</td>
<td>3.1e-2</td>
<td>3.1e3</td>
<td>47</td>
<td>6.0e4</td>
</tr>
<tr>
<td>100</td>
<td>6.0e-3</td>
<td>3.0</td>
<td>6.4</td>
<td>2.4e3</td>
<td>8.6e-4</td>
<td>1.1e3</td>
<td>4.8</td>
<td>9.1e4</td>
</tr>
<tr>
<td>140</td>
<td>1.1e-3</td>
<td>3.0</td>
<td>1.1</td>
<td>3.3e3</td>
<td>2.1e-5</td>
<td>7.6e2</td>
<td>0.98</td>
<td>4.9e5</td>
</tr>
<tr>
<td>180</td>
<td>2.5e-4</td>
<td>3.2</td>
<td>0.17</td>
<td>2.1e3</td>
<td>7.8e-7</td>
<td>1.3e2</td>
<td>0.15</td>
<td>9.9e5</td>
</tr>
</tbody>
</table>

Table 5.6: 6D parabolic thermal fin problem: convergence rate and effectivities of traditional reduced basis
5.3 Algorithm performance

The optimization problem is solved using the FE-only interior point, traditional RB interior point, and trust region RB approaches. We consider ten random least-squares cost functions (as discussed in Section 5.1.3) and solve the optimization for each cost function using the same set of ten random initial conditions, resulting in a total of 100 optimization trials. Algorithm parameters and problem data used are tabulated in Table 5.1 and Table 5.2. Performance results measured in terms of run time and required number of FE evaluations for the elliptic case are presented in Figures 5-2 and 5-3, and in Figures 5-4 and 5-5 for the parabolic case. Note that in the parabolic case the number of FE evaluations stated in the figures corresponds to the number of full forward integrations in time.

Overall, the combined trust region reduced basis optimization approach consistently reduces the number of FE evaluations required to locate an optimum relative to the two other approaches tested. On average, compared to the FE-only method, the trust region RB approach requires 39% (30%) as many full solves in the 2D (6D) elliptic case. In the parabolic case, the trust region RB approach requires 30% (14%) as many full solves in the 2D (6D) case. We also note that in our numerical trials, Line 12 of Algorithm 2 is never reached — i.e., we never 'waste' a full solve by updating the model (Line 8) at a potential iterate only to reject it. Although we cannot guarantee this behavior in general, by defining the trust region systematically via the error estimates, we can influence how often this would occur.

By building the RB adaptively along the optimization trajectory, the trust region reduced basis method also reduces the number of full evaluations needed relative to the number of full solves needed in the offline phase of the traditional RB approach: on average, the trust region RB requires 37% and 12% of the full solves needed for the 2D and 6D elliptic cases, and 25% and 3% of the full solves required in the 2D and 6D parabolic cases. Finally, the trust region RB approach is able to reduce the number of evaluations of the reduced system relative to the offline phase of the traditional RB approach, by a mean factor of 3 (3000) in the 2D (6D) elliptic case, and by a mean
factor of 6 (8600) in the 2D (6D) parabolic case.

Figure 5-2: Runtime comparison for optimizations constrained by elliptic PDEs. In contrast, the traditional offline-online reduced-basis approach for a 2D (6D) optimization runs in 0.04 (0.10) seconds online, but requires 1.6 (4800) seconds offline (on average).

Figure 5-3: Number of full model evaluations required for optimizations constrained by elliptic PDEs. The traditional offline-online RB approach requires 0 full evaluations online, and an average of 9 (48) full evaluations in the 2D (6D) case offline.

Whether or not the achieved reduction in number of full FE solves translates into a run time speedup depends on the size and complexity of the problem. The trust region RB run time averages 83% of the FE-interior point run time in the elliptic 2D case, 82% in the elliptic 6D case, and 72% and 44% in the parabolic 2D and 6D cases, respectively. Gains in the elliptic case are small because the full elliptic problem can be solved inexpensively. In the parabolic 2D case, the overhead involved in building
the reduced-basis fills most of the time saved by the three-fold reduction in full solves. In contrast, in the parabolic 6D optimization, the seven-fold reduction in full solves makes the RB overhead a much smaller portion of the overall optimization time. The offline computational investment in the traditional RB approach makes it slower than both our proposed approach and the FE-only approach for all cases tested.

Figure 5-4: Runtime comparison for optimizations constrained by parabolic PDEs. For comparison, the average traditional offline-online RB 2D (6D) optimization runs in 0.23 (2.7) seconds online, but requires 125 seconds (60 hours) offline.

Figure 5-5: Number of full model evaluations required for optimizations constrained by parabolic PDEs. The traditional offline-online RB approach requires 0 full evaluations online, and an average of 15 (151) full evaluations in the 2D (6D) case offline.

We summarize the achieved gains in Table 5.7. Our results suggest that there may be potential for greater savings in optimizations of higher parameter dimension.
Additionally, we note that the size of our FE discretization is fairly small, especially relative to the size of FE discretizations that might be encountered in real-world problems. Because the trust region approach is able to significantly reduce the number of FE solves relative to the FE-only approach, there is also potential for greater gains in problems with higher-dimensional discretizations.

<table>
<thead>
<tr>
<th></th>
<th>Trust region RB run time</th>
<th>Trust region RB # FE solves</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FE-fmincon run time</td>
<td>FE-fmincon # FE solves</td>
</tr>
<tr>
<td></td>
<td>min  mean  max</td>
<td>min  mean  max</td>
</tr>
<tr>
<td>2D Elliptic</td>
<td>0.20 0.83 1.48</td>
<td>0.09 0.39 0.75</td>
</tr>
<tr>
<td>6D Elliptic</td>
<td>0.31 0.82 2.32</td>
<td>0.13 0.30 0.70</td>
</tr>
<tr>
<td>2D Parabolic</td>
<td>0.38 0.72 1.32</td>
<td>0.20 0.30 0.44</td>
</tr>
<tr>
<td>6D Parabolic</td>
<td>0.21 0.44 0.80</td>
<td>0.10 0.14 0.24</td>
</tr>
</tbody>
</table>

Table 5.7: Summary of gains made by proposed trust region reduced basis optimization approach relative to MATLAB fmincon. Time gains are seen in the majority of test cases, and a reduction in the number of required FE function evaluations is seen in all cases tested, with the largest reduction in the 6D parabolic optimization.
Chapter 6

Conclusions

We have introduced a combined reduced basis trust region framework for PDE-constrained optimization of quadratic cost functionals, as well as novel \textit{a posteriori} error bounds for the reduced basis cost approximation and its gradient. In this approach, reduced basis models are leveraged in several ways: First, reduced basis models are used as the model function within the trust region optimization, reducing the time for each optimization function evaluation. Second, the reduced basis \textit{a posteriori} error bounds are used to choose when to accept and reject trust region optimization iterates. Third, the error bounds are used to systematically determine when to update the reduced model. Fourth and finally, the existence of error bounds for RBM allows rigorous proof of convergence of the algorithm to a stationary point of the full model. We have implemented the proposed algorithm on a thermal fin model problem using least-squares cost functions with up to 6 variable parameters, and achieve reductions in the number of full evaluations needed relative to a high-fidelity interior point approach in all cases tested, with up to 3-fold gains in the elliptic case and 7-fold gains in the parabolic case.
Bibliography


